IBM WebSphere Information Integrator
OmniFind Edition

Integration der Textanalyse

Version 8.3

SC12-3611-00

IBM WebSphere Information Integrator
OmniFind Edition

Integration der Textanalyse

Version 8.3

SC12-3611-00

Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten Sie die allgemeinen Informationen unter ″Bemerkungen″ lesen.

Sie können IBM Veröffentlichungen online oder über Ihren lokalen IBM Ansprechpartner bestellen.

- Veröffentlichungen bestellen Sie online über das IBM Publications Center unter der Internetadresse www.ibm.com/shop/publications/order.
- Ihren lokalen IBM Ansprechpartner finden Sie über die Internetsite ″IBM Directory of Worldwide Contacts″ unter der Internetadresse www.ibm.com/planetwide.

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
IBM WebSpere Information Integrator OmniFind Edition, Text Analysis Integration,
IBM Form SC18-9674-00,
herausgegeben von International Business Machines Corporation, USA

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderung des Textes bleibt vorbehalten.

Herausgegeben von:
SW TSC Germany
Kst. 2877
November 2005

Inhaltsverzeichnis

Informationen zu diesen Themen

Verwenden Sie diese Informationen, um Lösungen für die semantische Suche in einem IBM WebSphere Information Integrator OmniFind Edition-System der Version 8.3 zu erstellen und zu implementieren. Die semantische Suche ermöglicht es Ihnen, nach Konzepten auf höheren Ebenen zu suchen und Beziehungen in einer Suchabfrage auszudrücken, die mit Hilfe einer Textanalyse festgestellt werden.

WebSphere Information Integrator OmniFind Edition (WebSphere II OmniFind Edition) stellt eine Technologie bereit, die *Unternehmenssuche* genannt wird. Die Komponenten für die Unternehmenssuche werden bei der Installation von WebSphere II OmniFind Edition installiert. Der Begriff *Unternehmenssuche* wird in der gesamten Dokumentation zu WebSphere II OmniFind Edition verwendet, mit Ausnahme von Verweisen auf Installationspfade und Bezeichnungen von Produktpaketen.

In der Dokumentation zur Textanalyse für die Unternehmenssuche werden die folgenden Themen behandelt:

- Eine Einführung in die linguistische Unterstützung im Unternehmen
- Anweisungen, wie eine benutzerdefinierte Textanalyse in die Unternehmenssuche integriert wird
- Anweisungen, wie XML-Dokumentstrukturen zugeordnet werden
- Anweisungen, wie JDBC-Tabellen ausgewählte Analyseergebnisse hinzugefügt werden
- Anweisungen, wie dem Index für die Unternehmenssuche Analyseergebnisse hinzugefügt werden, um die semantische Suche zu aktivieren
- Anweisungen, wie Verzeichnisse von Synonymen, Stoppwörtern und Boostwörtern in die Suche eingeschlossen werden
- Eine Übersicht darüber, welche Textanalysefunktionalität automatisch während der Dokumentbearbeitung ausgeführt wird

Zielgruppe

Diese Informationen richten sich an Systemadministratoren und Entwickler von Suchanwendungen, die für das Erstellen und Implementieren von Lösungen für die semantische Suche in der Unternehmenssuche verantwortlich sind.

Die Unternehmenssuche unterstützt die semantische Suche in Textdokumenten. Dokumente werden analysiert, und die Ergebnisse werden gespeichert. Während der Suche kann auf diese Ergebnisse zugegriffen werden. Eine Kombination aus Textanalyse und Unternehmenssuche, die in der Lage ist, sowohl Wörter als auch Textauszüge zu indexieren, ermöglicht die semantische Suche. Die Unterstützung der semantischen Suche während des Suchens hat zum Ziel, die Ergebnisse der Dokumentsuche zu verbessern, das heißt, die bestmögliche Auswahl an Dokumenten bereitzustellen, die mit der Abfrage übereinstimmen.

Mit Hilfe dieser Informationen lernen Sie, wie Sie eine benutzerdefinierte Textanalyse in die Unternehmenssuche integrieren und wie Sie Verzeichnisse von Synonymen, Stoppwörtern und Boostwörtern verwenden, um Ihre Abfrageergebnisse zu verbessern. Weiterhin werden Sie anhand dieser Informationen verstehen, welche grundlegende linguistische Unterstützung in der Unternehmenssuche ständig während der Dokumentbearbeitung enthalten ist.

Zur effektiven Nutzung dieser Informationen müssen Sie mit Webanwendungen vertraut sein und über Erfahrung mit den Datenquellen verfügen, die Sie durchsuchen möchten.

Linguistische Unterstützung der semantischen Suche

Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, wie z. B. Japanisch.

Die linguistische Unterstützung während der Suche hat zum Ziel, die Ergebnisse der Dokumentsuche zu verbessern und die bestmögliche Auswahl an Dokumenten bereitzustellen, die mit der Abfrage übereinstimmen.

Die Verarbeitung auf linguistischer Basis erfolgt in zwei Arbeitsabschnitten: Wenn ein Dokument verarbeitet wird, damit es dem Index hinzugefügt werden kann, und wenn ein Benutzer während einer Suche eine Abfrage eingibt.

Die Unternehmenssuche enthält nur eine grobe linguistische Funktionalität, die erforderlich ist, um die Sprache eines Eingabedokuments zu ermitteln und den Eingabedatenstrom des Dokuments in Wörter oder Token zu segmentieren.

Wenn Sie wissen, dass Sie sich hauptsächlich auf eine Suche beschränken werden, die die Dokumentstruktur verwendet, wie z. B. die Suche nach Basisschlüsselwörtern oder nach nativer XML, werden Ihre Anforderungen an die Suche von der in der Unternehmenssuche enthaltenen Verarbeitung auf linguistischer Basis in ausreichendem Maße erfüllt.

Die Verarbeitung auf linguistischer Basis allein ist jedoch nicht immer ausreichend, wenn Sie spezifischer suchen und über die einzelnen, im Dokument enthaltenen Wörter hinausgehen wollen. Dies wird in den folgenden Beispielen verdeutlicht:

- Bei der Onlinezusammenarbeit sind Informationen nicht immer explizit gekennzeichnet, zum Beispiel wenn eine Adresse oder Telefonnummer in einer E-Mail angegeben wird. Der Begriff *Telefonnummer* wird nicht verwendet. Statt dessen enthält die E-Mail einen Satz wie "Sie erreichen mich unter 0711/24798".
- In der marktorientierten Informationsbeschaffung werden in Dokumenten Mitbewerber und die von ihnen gelieferten Waren erwähnt, oder die Website Ihres Mitbewerbers wechselte innerhalb der letzten Monate von einer Produktgruppe zu einer anderen.
- Im Customer-Relationship-Management werden möglicherweise Probleme an Bremsen von Autos in Reparaturwerkstätten im Einzugsgebiet von Köln erwähnt. In den Berichten der Reparaturwerkstatt wird das Problem wie folgt beschrieben: "Bremsbacke justiert wegen auslaufender Hydraulik". Darüber hinaus wird in den Berichten nur der Name der Straße angegeben, in der sich die Werkstatt befindet, nicht die vollständige Adresse.
- In der Forschung wird in Dokumenten ein bestimmtes Protein beschrieben und seine Beziehung zu mindestens einer Krankheit, die im selben Abschnitt genannt wird. In der Literatur gibt es mehr als zwanzig Namen für dieses Protein und häufig wird in den Dokumenten das Wort *Krankheit* gar nicht erwähnt, sondern nur der Name der Krankheit.

In diesen Beispielen werden Sie bei der Suche nach Ihren gewünschten Informationen in den umfangreichen Sammlungen an Informationsquellen, die es heute gibt, vor neue Herausforderungen gestellt, für die eine ausgereifte Analyse erforderlich ist, die die in der Unternehmenssuche angebotene Segmentierungsstufe und auf Basis von Wörterverzeichnissen ausgeführte Analyse übersteigt. Ein großer Teil der

gesuchten Informationen ist in den Originaldokumenten nicht ausdrücklich gekennzeichnet oder hervorgehoben. Statt dessen müssen die Informationen analysiert werden, damit die betreffenden Konzepte erkannt und gefunden werden, z. B. benannte Entitäten wie Personen, Organisationen, Standorte, Einsatzmittel und Produkte und die möglichen zwischen diesen Entitäten bestehenden Beziehungen.

IBM Unstructured Information Management Architecture (UIMA) ist eine Architektur und Software-Rahmendefinition, mit deren Hilfe Sie die erweiterte Analysefunktionalität, die Sie brauchen, um die von Ihnen gesuchten Informationen in Dokumentobjektgruppen zu erkennen und zu finden, in der Unternehmenssuche erstellen können.

Zugehörige Konzepte

„Integration der benutzerdefinierten Textanalyse" auf Seite 3
Unstructured Information Management Architecture (UIMA) ist eine Softwarearchitektur, die das Erstellen, Aufspüren und Einsetzen der Textanalysefunktionalität unterstützt. Mit UIMA können Sie eine benutzerdefinierte Textanalyse erstellen.

„Unstructured Information Management Architecture (UIMA) - Übersicht" auf Seite 4
Unstructured Information Management Architecture (UIMA) ist eine Architektur und Software-Rahmendefinition, mit deren Hilfe Sie die erweiterte Analysefunktionalität erstellen können, um damit nach bestimmten Informationen in Dokumentobjektgruppen zu suchen.

Integration der benutzerdefinierten Textanalyse

Unstructured Information Management Architecture (UIMA) ist eine Softwarearchitektur, die das Erstellen, Aufspüren und Einsetzen der Textanalysefunktionalität unterstützt. Mit UIMA können Sie eine benutzerdefinierte Textanalyse erstellen.

UIMA ist eine offene Plattform, die für jede Analysefunktion mit eindeutigem Konzept Komponenten angibt und sicherstellt, dass diese Komponenten ohne großen Aufwand wiederverwendet und miteinander kombiniert werden können.

Ein zentrales Konzept von UIMA ist die *Analysesteuerkomponente*, die für das Aufspüren und Darstellen von Analyseinhalten in Textdokumenten verantwortlich ist. Die logische Analysekomponente wird *Annotator* genannt. Ein Annotator führt nur eine Analysetask aus und befasst sich nicht mit der übrigen Verarbeitung. Eine Analysesteuerkomponente kann einen einzelnen Annotator enthalten, sie kann aber auch aus mehreren Steuerkomponenten zusammengesetzt sein, von denen jede Annotatoren enthält.

Die von einer Analysesteuerkomponente generierten Informationen werden als *Analyseergebnisse* bezeichnet. Im Idealfall entsprechen die Analyseergebnisse den Informationen, nach denen Sie suchen wollen.

Die erweiterte linguistische Analyse enthält eine Kombination aus vielen verschiedenen Analysetasks. Die Analyse beginnt z. B. bei der Erkennung der Sprache und der Segmentierung, auf die eine Wortarterkennung und anschließend eine tiefgehende grammatikalische Syntaxanalyse folgt. Der letzte Schritt besteht dann z. B. darin, dass der Zusammenhang zwischen bestimmten chemischen Substanzen und dem Auftreten bestimmter Symptome erkannt wird. Jeder Schritt im Analyseprozess ist für den nachfolgenden Schritt erforderlich.

UIMA stellt die Grundbausteine bereit, mit denen Sie Ihre eigenen Analysesteuerkomponenten erstellen, testen und implementieren können. Es stellt Ihnen jedoch keine linguistische Analysefunktionalität in Form von vorkonfigurierten Analysesteuerkomponenten zur Verfügung, die Sie in Ihrer UIMA-Umgebung implementieren können.

UIMA Software Development Kit (SDK) enthält eine Java-Implementierung der UIMA-Rahmendefinition für die Implementierung, die Beschreibung, die Zusammensetzung und den Einsatz von UIMA-Komponenten. Weiterhin stellt es eine Eclipse-basierte Entwicklungsumgebung bereit, die eine Reihe von Tools und Dienstprogrammen für die Verwendung von UIMA enthält. Informationen zu Eclipse finden Sie unter www.eclipse.org.

Wenn Sie mit UIMA arbeiten wollen, müssen Sie UIMA Software Development Kit installieren. Das Development-Kit ist auf der Site von IBM developerWorks verfügbar. Weitere Informationen finden Sie im WebSphere Information Integrator-Bereich unter http://www.ibm.com/developerworks/db2/zones/db2ii/. In der Dokumentation von UIMA finden Sie Anweisungen zur Installation von UIMA Software Development Kit in der interaktiven Eclipse-Entwicklungsumgebung.

Zugehörige Konzepte

„Linguistische Unterstützung der semantischen Suche” auf Seite 1
Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, wie z. B. Japanisch.

„Unstructured Information Management Architecture (UIMA) - Übersicht”
Unstructured Information Management Architecture (UIMA) ist eine Architektur und Software-Rahmendefinition, mit deren Hilfe Sie die erweiterte Analysefunktionalität erstellen können, um damit nach bestimmten Informationen in Dokumentobjektgruppen zu suchen.

Unstructured Information Management Architecture (UIMA) - Übersicht

Unstructured Information Management Architecture (UIMA) ist eine Architektur und Software-Rahmendefinition, mit deren Hilfe Sie die erweiterte Analysefunktionalität erstellen können, um damit nach bestimmten Informationen in Dokumentobjektgruppen zu suchen.

Eine *Komponentenstruktur* ist die zu Grunde liegende Datenstruktur, die das Ergebnis einer Analyse darstellt. Eine Komponentenstruktur ist eine Attribut-Wert-Struktur. Jede Komponentenstruktur hat einen Typ, und jeder Typ hat eine bestimmte Menge gültiger Komponenten oder Attribute (Merkmale), ähnlich wie eine Java-Klasse. Komponenten haben einen Bereichstyp, der den Wertetyp angibt, den die Komponente aufweisen muss, zum Beispiel Zeichenfolge.

Die meisten Analysealgorithmen, auch Annotatoren genannt, stellen ihre Analyseergebnisse in der Form von Annotationen bereit. Annotationen sind eine besondere Art Komponentenstruktur, die für die linguistische Analyseverarbeitung bestimmt ist. Eine Komponentenstruktur umfasst einen Teil des Eingabetexts und wird mit Hilfe von Anfangs- und Endpositionen im Eingabetext definiert.

So erstellt z. B. ein Annotator, der monetäre Ausdrücke erkennt, für den Text ″100,55 US Dollar″ eine Annotation des Typs `monetaryExpression`, der den Text mit der auf ″$″ gesetzten Komponente `currencySymbol` erfasst.

Alle Annotatoren in UIMA verwenden Komponentenstrukturen, um Informationen zu speichern oder zu lesen. In anderen Worten, alle Daten werden in Komponentenstrukturen modelliert.

Das Typsystem definiert alle möglichen Komponentenstrukturen nach Typen und Komponenten, ähnlich wie eine Klassenhierarchie in Java.

Alle Komponentenstrukturen werden in einer zentralen Datenstruktur dargestellt, die als die *allgemeine Analysestruktur* bezeichnet wird. Der gesamte Datenaustausch erfolgt unter Verwendung der allgemeinen Analysestruktur.

Die allgemeine Analysestruktur enthält die folgenden Objekte:

- Das Textdokument
- Die Beschreibung des Typsystems, in der die Typen, Subtypen und ihre Komponenten angegeben sind
- Analyseergebnisse, die das Dokument oder einen Dokumentbereich beschreiben
- Ein Indexrepository, das den Zugriff auf und die Iteration für die Analyseergebnisse unterstützt

Zugehörige Konzepte

„Linguistische Unterstützung der semantischen Suche” auf Seite 1
Die Unternehmenssuche bietet die linguistische Unterstützung der Suche für Textdokumente in den meisten indo-europäischen und asiatischen Sprachen an, wie z. B. Japanisch.

„Integration der benutzerdefinierten Textanalyse” auf Seite 3
Unstructured Information Management Architecture (UIMA) ist eine Softwarearchitektur, die das Erstellen, Aufspüren und Einsetzen der Textanalysefunktionalität unterstützt. Mit UIMA können Sie eine benutzerdefinierte Textanalyse erstellen.

Workflow für die Integration der benutzerdefinierten Analyse

Algorithmen für die benutzerdefinierte Textanalyse werden mit UIMA Software Development Kit erstellt und getestet, anschließend in die Unternehmenssuche implementiert und für Dokumentobjektgruppen ausgeführt.

Gehen Sie wie folgt vor, um Analysealgorithmen zu entwickeln und in die Unternehmenssuche zu implementieren:

1. Planung und Design:
   a. Legen Sie fest, nach welchen Informationen Sie suchen wollen. Welche Dokumente wollen Sie abrufen? Welche Konzepte und Beziehungen sind in einer bestimmten Suchtask erforderlich? So können z. B. die Namen von Produkten und Mitarbeitern erforderlich sein, um die allgemeine Suche auf der internen Website eines pharmazeutischen Unternehmens zu erweitern, während für den Bereich Forschung und Entwicklung Varianten von Medikamentennamen und die Beziehungen zwischen Medikament, Ursache und Heilung erforderlich sind.
   b. Geben Sie die Art Textanalyse an, die Sie benötigen, um die Informationen aus den zu durchsuchenden Dokumenten abzurufen.
   c. Wenn Ihre Objektgruppe XML-Dokumente enthält, müssen Sie entscheiden, ob Sie XML-Markup in Ihrer Lösung verwenden wollen. In der Unternehmenssuche gibt es zwei Möglichkeiten, wie Sie XML-Markup verwenden können:
      - Wenn Sie XML-Markup in Ihrer benutzerdefinierten Analyse verwenden können (z. B. wenn Ihre Dokumente die Elemente `<summary>` oder `<topic>` enthalten, die in einem Zusammenfassungs- oder Kategorisierungsannotator nützlich sein können), definieren Sie Zuordnungen zwischen XML und der allgemeinen Analysestruktur.
      - Wenn Sie XML-Markup in Ihren Abfragen so verwenden wollen, wie es im Dokument angezeigt wird, aktivieren Sie die native XML-Zuordnung.
   d. Legen Sie fest, auf welche Textanalyseergebnisse, die in der allgemeinen Analysestruktur gespeichert sind, Sie über die semantische Suche zugreifen wollen. Definieren Sie die Zuordnung zwischen der allgemeinen Analysestruktur und dem Index.
   e. Legen Sie fest, ob Sie Analyseergebnisse in einer relationalen Datenbank speichern wollen, z. B. um Trends und Beziehungen aufzuspüren, indem Sie Berichterstellungs- oder Data-Mining-Anwendungen verwenden. Definieren Sie die Zuordnung zwischen der allgemeinen Analysestruktur und JDBC-Tabellen.
   f. Entwerfen Sie die semantische Suchanwendung. Legen Sie fest, wie der Benutzer der Suche die zusätzliche semantische Suchfunktionalität verwenden wird. Entwerfen Sie die Benutzeroberfläche.
2. Entwicklung: Aktivitäten mit UIMA Software Development Kit

a. Definieren Sie die einzelnen Analyseschritte.
b. Beschreiben Sie das Typsystem für Ihre Zuordnungen und Analysealgorithmen.
c. Entwickeln Sie die Analysealgorithmen (Annotatoren) für jeden Analyseschritt, und betten Sie die Annotatoren in Analysesteuerkomponenten ein, indem Sie UIMA Software Development Kit verwenden. Erstellen Sie alle benutzerdefinierten Analysen unter Verwendung der Basisfunktionalität (Spracherkennung und Einteilung in Token) im Annotatorpaket für die Unternehmenssuche.
d. Nachdem Sie die Analysealgorithmen in UIMA getestet haben, packen Sie die Analysesteuerkomponente als PEAR-Datei in ein Verarbeitungsenginearchiv. Das Archiv darf nur Ihre Analysealgorithmen enthalten, nicht jedoch die linguistische Basisfunktionalität für die Unternehmenssuche.

3. Implementierung: Aktivitäten für die Unternehmenssuche
   a. Übertragen Sie die Archivierungsdatei der Analysesteuerkomponente (.pear) an die Unternehmenssuche. Geben Sie der Analysekomponente einen Namen, damit Sie diesen in der Unternehmenssuche verwenden können.
   b. Verknüpfen Sie mindestens eine Dokumentobjektgruppe mit Ihrer Analysesteuerkomponente.
   c. Übertragen Sie gegebenenfalls die Zuordnungskonfiguration zwischen dem XML-Element und dem UIMA-Typ, die Sie für Ihre benutzdefinierte Analyse definiert haben, an jede Objektgruppe, und wählen Sie diese aus.
   d. Übertragen Sie gegebenenfalls die Datenbankzuordnungskonfiguration, die Sie für Ihre benutzerdefinierte Analyse definiert haben, an jede Objektgruppe, und wählen Sie diese aus.
   e. Übertragen Sie die Indexzuordnungskonfiguration, die Sie für die semantische Suche definiert haben, an jede Objektgruppe, und wählen Sie diese aus.
   f. Konfigurieren Sie gegebenenfalls Ihre benutzerdefinierte semantische Suchanwendung. Implementieren Sie z. B. Ihre browserbasierte Benutzeroberfläche zum Suchen in einem Anwendungsserver.
   g. Führen Sie für die Dokumente in Ihrer semantischen Suchobjektgruppe, wie für eine stichwortbasierte Objektgruppe, eine Crawlersuche, eine syntaktische Analyse und eine Indexierung aus.

Zugehörige Tasks

„Installieren und Ausführen der Basisannotatoren für die Unternehmenssuche" Sie können das Basisannotatorpaket für die Unternehmenssuche verwenden, um neue Annotatoren zu entwickeln, die auf der Ausgabe der Annotatoren für die Unternehmenssuche basieren, und um benutzerdefinierte Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu testen.

Installieren und Ausführen der Basisannotatoren für die Unternehmenssuche

Sie können das Basisannotatorpaket für die Unternehmenssuche verwenden, um neue Annotatoren zu entwickeln, die auf der Ausgabe der Annotatoren für die Unternehmenssuche basieren, und um benutzerdefinierte Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu testen.

Die Gruppe der Basisannotatoren umfasst folgende Elemente:

- **Sprach-ID-Annotator**

Ermittelt die Sprache eines Dokuments. Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jlangid.xml`.

- **Annotator für FROST-Wörterverzeichnissuche**

  Stellt Einteilung in Token und Satzerkennung auf der Basis der IBM LanguageWare-Wörterverzeichnisse bereit. Für Token werden zusätzliche linguistische Informationen generiert, z. B. Grundform oder Lemma. Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jfrost.xml`.

- **Leerzeichentokenizer**

  Führt eine leerzeichenbasierte Einteilung in Token für alle in europäischen Sprachen abgefassten Dokumente oder andere durch Leerzeichen getrennte Scripts aus. Darüber hinaus kann der Annotator eine Einteilung in Token mit Hilfe von n-gram-Elementen für die folgenden Textscripts ausführen: Arabisch, Han, Hebräisch, Hiragana, Katakana, Lao, Mongolisch, Thailändisch, YI und Hangul. Informationen zu Funktionalität und Konfigurationsparametern finden Sie in der Deskriptordatei `jtok.xml`.

Damit Sie diese Annotatoren in UIMA ausführen können, muss UIMA Software Development Kit (SDK) installiert sein. Es ist auf der Website von IBM developerWorks unter http://www-128.ibm.com/developerworks/db2/zones/db2ii/ verfügbar.

Das Basisannotatorpaket für die Unternehmenssuche ist eine komprimierte Datei, die die in der Unternehmenssuche verwendeten Annotatoren für die Textanalyse enthält. Diese Annotatoren werden stets vor einer benutzerdefinierten Analyse ausgeführt, wenn in der Unternehmenssuche Dokumente syntaktisch analysiert werden.

Gehen Sie wie folgt vor, um das Annotatorpaket zu installieren:

1. Suchen Sie in Ihrer Installation der Unternehmenssuche (WebSphere Information Integrator OmniFind Edition) im Verzeichnis *ES_INSTALL_ROOT*`/packages/uima` nach dem Annotatorpaket `OF_base_annotators.zip`.
2. Kopieren Sie die komprimierte Datei in das Stammverzeichnis Ihrer UIMA SDK-Installation.
3. Extrahieren Sie die komprimierte Datei, um die Basisannotatordateien für die Unternehmenssuche der angegebenen Verzeichnisstruktur Ihrer UIMA SDK-Installation hinzuzufügen.

Nachdem Sie das Basisannotatorpaket installiert haben, finden Sie die Deskriptordateien im Ordner *UIMA_SDK_INSTALL*`/docs/examples/descriptors/analysis_engine`. Die Datei `of_tokenization.xml` listet die Basisannotatoren in der Reihenfolge auf, in der sie in der Unternehmenssuche verwendet werden.

Die Deskriptordateien enthalten dieselben Konfigurationswerte, die in der Unternehmenssuche verwendet werden. Sie können Werte zu Debugging-Zwecken in UIMA SDK ändern. Sie sollten jedoch nicht die Deskriptordateien in Ihrem System für die Unternehmenssuche ändern. Änderungen an diesen Dateien können zu Systeminstabilität oder Leistungsproblemen führen.

Das Basisannotatorpaket für die Unternehmenssuche enthält nur die Wörterverzeichnisse, die für die Verarbeitung englischsprachiger Dokumente erforderlich sind. Wenn Sie andere Sprachen in Ihrer Entwicklungsumgebung verarbeiten wollen, gehen Sie wie folgt vor:

1. Suchen Sie in der Installation Ihrer Unternehmenssuche unter *ES_INSTALL_ROOT*/configurations/parserservice/jediidata/frost/resources nach den Wörterverzeichnissen für die Unternehmenssuche.
2. Kopieren Sie den Inhalt des Verzeichnisses in Ihre lokale Installation von UIMA SDK unter *UIMA_SDK_INSTALL*/data/frost/resources.

Gehen Sie wie folgt vor, um zu prüfen, ob das Annotatorpaket erfolgreich installiert wurde:

1. Öffnen Sie CAS-Visual Debugger (CVD) im folgenden Verzeichnis: *UIMA_SDK_INSTALL*/bin/cvd[.bat/.sh].
2. Klicken Sie **Run → load TAE** an.
3. Wählen Sie die Kennungsdatei of_tokenization.xml für die Textanalysesteuerkomponente im Verzeichnis *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine aus.
4. Laden Sie ein Beispieldokument und führen Sie die Textanalysesteuerkomponente aus. In CVD werden Annotationen des Typs uima.tt.TokenAnnotation angezeigt.

Gehen Sie wie folgt vor, um die Annotatoren für die Unternehmenssuche für Ihre Verarbeitung zu verwenden:

1. Fügen Sie dem Abschnitt typeSystem Ihrer benutzerdefinierten Annotatorkennung einen Verweis auf die Datei of_typesystem.xml hinzu, wenn Ihr benutzerdefinierter Annotator Typen verwendet, die von Annotatoren für die Unternehmenssuche definiert werden. Die Datei of_typesystem.xml befindet sich im Verzeichnis *UIMA_SDK_INSTALL*/docs/examples/descriptors/analysis_engine. In der Datei jtok.xml im Verzeichnis analysis_engine finden Sie ein Beispiel, wie einer Deskriptordatei Verweise hinzugefügt werden.

   Zugehörige Verweise

   „In der Unternehmenssuche definierte Typen und Komponenten" auf Seite 46
   Das in der Unternehmenssuche definierte Typsystem umfasst die Handhabung von Metadaten und eine linguistische Basisanalyse.

Textanalysealgorithmen

UIMA Software Development Kit enthält APIs und Tools, mit denen Sie Annotatoren (Analysealgorithmen, einschließlich der Typsystembeschreibung) erstellen und in Analysesteuerkomponenten einbetten können.

Die Dokumentation von UIMA enthält einen Leitfaden im Stil eines Lernprogramms, mit dessen Hilfe Sie diese Komponenten erstellen können. Das Software-Development-Kit enthält Dienstprogramme zum Testen und Anzeigen Ihrer Ergebnisse und eine kleine semantische Suchmaschine zum Indexieren Ihrer Analyseergebnisse. Sie können auch eine erweiterte Suche für die im Index gespeicherten Informationen ausführen.

UIMA Software Development Kit stellt keine vorkonfigurierten Analysesteuerkomponenten zur Verfügung. Sie können jedoch die Basisannotatoren in einer UIMA-Umgebung verwenden, die in der Unternehmenssuche bereitgestellt werden. In der Dokumentation von UIMA finden Sie Informationen zum Implementie-

ren der Funktionalität für Spracherkennung und Einteilung in Token vor den Textanalysealgorithmen, wenn Sie diese in Ihrer UIMA-Umgebung entwickeln.

Nachdem Sie Ihre Analysesteuerkomponenten unter Verwendung von UIMA Software Development Kit entwickelt und getestet haben und wenn Sie diese Algorithmen für eine Dokumentobjektgruppe in der Unternehmenssuche ausführen wollen, müssen Sie eine PEAR-Datei erstellen. Diese Archivierungsdatei enthält alle erforderlichen Ressourcen für die Implementierung Ihrer benutzerdefinierten Analysefunktionalität als Analysesteuerkomponenten für die Unternehmenssuche. Alle Verarbeitungsschritte, die zum Erstellen eines Archivs erforderlich sind, sind in der Dokumentation von UIMA beschrieben, die im Software-Development-Kit bereitgestellt wird.

Das Archiv darf nur Ihre benutzerdefinierte Analyse enthalten, selbst wenn diese auf der in der Unternehmenssuche bereitgestellten linguistischen Basisfunktionalität basiert. Die Basisanalyseschritte der Unternehmenssuche werden stets vor einer benutzerdefinierten Analyse ausgeführt.

Wenn Sie lernen wollen, wie Sie eine semantische Suchlösung für die Unternehmenssuche konfigurieren und implementieren können, führen Sie das Lernprogramm aus. Sie finden es unter
http://www.ibm.com/developerworks/db2/zones/db2ii/. Das Lernprogramm führt Sie durch die Schritte, die erforderlich sind, um benutzerdefinierte Textanalysealgorithmen für die Unternehmenssuche zu implementieren und zeigt Ihnen, wie Sie die Analyseergebnisse in Abfragen verwenden können, um die Suchergebnisse zu verbessern.

Zugehörige Tasks

„Installieren und Ausführen der Basisannotatoren für die Unternehmenssuche" auf Seite 6
Sie können das Basisannotatorpaket für die Unternehmenssuche verwenden, um neue Annotatoren zu entwickeln, die auf der Ausgabe der Annotatoren für die Unternehmenssuche basieren, und um benutzerdefinierte Annotatoren innerhalb von UIMA Software Development Kit (SDK) zu testen.

Typsystembeschreibung

Die Typsystembeschreibung enthält die Komponentenstrukturen (die zu Grunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Dieselben Typen, die in der Typsystembeschreibung definiert sind, müssen von der Analysesteuerkomponente, die die Annotatoren (Analysealgorithmen) enthält, und in allen Zuordnungsdateien verwendet werden, die mit der benutzerdefinierten Analyse verknüpft sind. Hierbei kann es sich um die Konfigurationsdatei für die XML-Zuordnung, die Konfigurationsdatei für die Indexerstellung oder die JDBC-fähige Zuordnungsdatei für die Datenbankkonfiguration handeln.

Die Typsystembeschreibung eines Annotators kann Teil des Deskriptors des Annotators sein, oder sie kann in einer separaten Deskriptordatei für das Typsystem enthalten sein. Manchmal ist sie auch Teil des Deskriptors eines anderen Annotators, der in derselben Analysesteuerkomponente enthalten ist.

Die Typsystembeschreibung muss Teil des Analysesteuerkomponentenarchivs (PEAR-Datei) sein, die aus Ihrer UIMA-Umgebung in die Unternehmenssuche importiert wird.

Die folgende Beispielbeschreibung eines Typsystems wird in allen Themen verwendet, in denen die unterschiedlichen Zuordnungstypen erläutert werden, die Sie mit der benutzerdefinierten Analyse auswählen können.

Das folgende Beispiel einer Typsystembeschreibung enthält Polizeiberichte mit Informationen zu Verdächtigen, Tatorten, Tatzeiten und Datumsangaben:

```
<?xml version="1.0" encoding="UTF-8"?>
<typeSystemDescription>
   <name>Police Reports Type System</name>
   <description>Typsystembeschreibung für
        Polizeiberichte</description>
   <version>1.0</version>
   <types>
     <typeDescription>
       <name>com.ibm.omnifind.types.PoliceReport</name>
       <description>Kommentiert einen Polizeibericht</description>
       <superTypeName>uima.tcas.Annotation</superTypeName>
       <features>
         <featureDescription>
           <name>time</name>
           <description>Zeit, zu der die Tat laut Aussage begangen wurde
              </description>
           <rangeTypeName>com.ibm.omnifind.types.Time</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>date</name>
           <description>Datum, an dem die Tat begangen wurde</description>
           <rangeTypeName>com.ibm.omnifind.types.Date</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>location</name>
           <description>Ort, an dem die Tat begangen wurde</description>
           <rangeTypeName>com.ibm.omnifind.types.City</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>knownSuspects</name>
           <description>Enthält Annotationen des Typs "Suspect" zu Verdächtigen</description>
           <rangeTypeName>uima.cas.FSArray</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>crimeDescription</name>
           <description>Kurzbeschreibung der Tat</description>
           <rangeTypeName>uima.cas.String</rangeTypeName>
         </featureDescription>
       </features>
     </typeDescription>
     <typeDescription>
       <name>com.ibm.omnifind.types.City</name>
       <description>Name einer Stadt</description>
       <superTypeName>uima.tcas.Annotation</superTypeName>
       <features>
         <featureDescription>
           <name>cityName</name>
           <description>Name der Stadt</description>
           <rangeTypeName>uima.cas.String</rangeTypeName>
         </featureDescription>
         <featureDescription>
           <name>cityDistrict</name>
           <description>Name des Stadtteils</description>
           <rangeTypeName>uima.cas.String</rangeTypeName>
         </featureDescription>
       </features>
     </typeDescription>
     <typeDescription>
       <name>com.ibm.omnifind.types.Person</name>
       <description>Annotation zu einer Person</description>
```

```
  <superTypeName>uima.tcas.Annotation</superTypeName>
  <features>
    <featureDescription>
      <name>role</name>
      <description>Rolle, z. B. Verdächtiger oder Zeuge</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>firstName</name>
      <description>Vorname der Person</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>surName</name>
      <description>Nachname der Person</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>title</name>
      <description>Anrede, z. B. Herr oder Frau</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>gender</name>
      <description>Männlich oder Weiblich</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Suspect</name>
  <description>Ermittelter Verdächtiger</description>
  <superTypeName>com.ibm.omnifind.types.Person</superTypeName>
  <features>
    <featureDescription>
      <name>description</name>
      <description>Beschreibung des Verdächtigen,
    z. B. Bartträger mit dunkler Brille</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
  </features>
</typeDescription>
<typeDescription>
  <name>com.ibm.omnifind.types.Date</name>
  <description>Datum</description>
  <superTypeName>uima.tcas.Annotation</superTypeName>
  <features>
    <featureDescription>
      <name>year</name>
      <description>Jahr, z. B. 2005</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>month</name>
      <description>Monat in Zahlen, z. B. 7</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>day</name>
      <description>Tag in Zahlen</description>
      <rangeTypeName>uima.cas.Integer</rangeTypeName>
    </featureDescription>
    <featureDescription>
      <name>dayOfWeek</name>
      <description>Wochentag, z. B. Montag</description>
      <rangeTypeName>uima.cas.String</rangeTypeName>
    </featureDescription>
```

```
        <featureDescription>
          <name>quarter</name>
          <description>Quartal, z. B. Q1-2005</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>englDate</name>
          <description>Datumsangabe im Format MM/TT/JJJJ</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
      </features>
    </typeDescription>
    <typeDescription>
      <name>com.ibm.omnifind.types.Time</name>
      <description>Zeit</description>
      <superTypeName>uima.tcas.Annotation</superTypeName>
      <features>
        <featureDescription>
          <name>hours</name>
          <description>Angabe der Stunde von 00-23</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>minutes</name>
          <description>Minuten der Stunde</description>
          <rangeTypeName>uima.cas.Integer</rangeTypeName>
        </featureDescription>
        <featureDescription>
          <name>timeOfDay</name>
          <description>Angabe der Tageszeit, wie Morgen oder Mittag</description>
          <rangeTypeName>uima.cas.String</rangeTypeName>
        </featureDescription>
      </features>
    </typeDescription>
  </types>
</typeSystemDescription>
```

Zugehörige Verweise

„In der Unternehmenssuche definierte Typen und Komponenten" auf Seite 46
Das in der Unternehmenssuche definierte Typsystem umfasst die Handhabung von Metadaten und eine linguistische Basisanalyse.

„In UIMA definierte Typen und Komponenten" auf Seite 49
UIMA Software Development Kit definiert linguistische Basistypen und -komponenten, die möglicherweise während der Textanalyse in einem Dokument festgestellt werden.

XML-Markup in Analyse und Suche

Sie können Informationen in XML-Strukturen in einem Dokument direkt einer allgemeinen Analysestruktur zuordnen, ohne einen UIMA-Annotator zu schreiben.

Wenn die Dokumente in Ihrer Objektgruppe in XML vorliegen und Sie XML-Markup in der Textanalyse oder semantischen Suche nutzen wollen, stehen Ihnen die folgenden Optionen zur Verfügung:

Native XML-Suche

Verwenden Sie diese Option, wenn Sie alle XML-Tags und -Attribute während der Suche so verwenden wollen, wie sie im Dokument angezeigt werden. Wenn Sie z. B. Dokumente für die Rechnungsstellung haben, die das Element `<addressee>` enthalten, können Sie durch Aktivieren der nativen XML-Suche diesen Tag in einer semantischen Suchabfrage verwenden, um in diesem Element nach einem bestimmten Kundennamen zu suchen.

Mit dieser Option wird die XML-Struktur des Dokuments in der allgemeinen Analysestruktur unter Verwendung des Typs `com.ibm.es.tt.MarkupTag` dargestellt. Für jeden XML-Tag wird eine Annotation dieses Typs erstellt. Die Annotation enthält den Namen des Tags, seine Attribute und den Attributinhalt. Diese Informationen werden immer indexiert und sind für die semantische Suche verfügbar.

Für die native XML-Suche ist keine Zuordnungskonfigurationsdatei erforderlich. Sie können die native XML-Suche über die Administrationskonsole für die Unternehmenssuche aktivieren.

Zuordnung eines XML-Elements zu einem UIMA-Typ

Verwenden Sie diese Option in den folgenden Fällen:

- Die Semantik bestimmter XML-Elemente ist präzise und kann in späteren Textanalyseschritten verwendet werden. Die Analyseschritte können direkt für die von den XML-Strukturen erstellten Annotationen und Komponenten ausgeführt werden und sind gegen die möglicherweise abweichenden Formate der Originaldokumente abgeschirmt. Z. B. enthält das Element `<addressee>` in Dokumenten für die Rechnungsstellung in der Regel Kundennamen. Wenn Sie die Zuordnung eines XML-Elements zu einem Typ verwenden, können Sie den Inhalt dieses Elements direkt Annotationen des Typs `Customer` zuordnen. Ein Annotator kann dann eine Beziehung zum Standort des Kunden ableiten, indem er die Informationen in der Umgebung der Annotation `Customer` verwendet.
- Sie wollen den Verarbeitungsbereich eines benutzerdefinierten Annotators auf bestimmte angegebene Bereiche in der XML-Eingabe eingrenzen. Sie wollen z. B. den Inhalt des Tags `<technicianComment>` in einem Annotator eingrenzen, der KFZ-Probleme feststellt.
- Sie wollen die Verarbeitung der Textanalyse und die nachfolgende Suche auf bestimmte Teile des XML-Dokuments beschränken und irrelevante Inhalte oder Inhalte herausfiltern, bei denen es sich nicht um Text handelt.
- Sie wollen XML-Tags, die unterschiedliche Namen haben (z. B. `<mainHeading>` oder `<doc>`) einem einheitlichen Bereich zuordnen, der bei der semantischen Suche verwendet wird (z. B. Titel).

In diesen Fällen, müssen Sie eine Konfigurationsdatei für die Zuordnung von XML zu UIMA-Typen erstellen, die die Komponentenstrukturen definiert. Die Komponentenstrukturen, die Sie in der Konfigurationsdatei definieren, werden während der syntaktischen Analyse der Dokumente erstellt. Der Zugriff erfolgt durch die benutzerdefinierten Annotatoren.

Sie können mehrere Konfigurationsdateien für eine Dokumentobjektgruppe verwenden. Welche Konfiguration für welches XML-Dokument verwendet wird, wird vom Element `<identifier>` festgelegt. Das Element `<identifier>` in der Konfigurationsdatei muss mit dem Stammelement im XML-Dokument übereinstimmen. Wenn das Stammelement Ihres Dokuments z. B. `doc` ist, muss in der Konfigurationsdatei für das Element `<identifier>` ebenfalls der Wert "doc" angegeben sein.

Wird keine Übereinstimmung gefunden, sucht das Programm nach einer Konfigurationsdatei, in der das Element `<identifier>` auf den Standardwert gesetzt ist. Wird keine Standardkonfiguration gefunden, werden die Textabschnitte des Dokuments (ohne Taginformationen) dem Dokumentkommentar in der allgemeinen Analysestruktur zugeordnet.

Wenn Sie Informationen extrahieren wollen, die nur in relevanten Teilen eines Dokuments enthalten sind, während die irrelevanten Teile ignoriert werden sollen, müssen Sie nur angeben, welche XML-Elemente des Dokuments relevante Informationen enthalten. Diese Vorgehensweise wird als Inhaltsextraktion bezeichnet. Sie können z. B. die in den Titel- und Hauptteilelementen angegebene Eingabe extrahieren, während Sie die Eingabe in den Elementen Autor, Datum, ID und Veröffentlichungskomponente ignorieren.

Die Inhaltsextraktion kann die Analyseverarbeitung für die folgenden XML-Dokumenttypen verbessern:

- Dokumente, deren Inhalt umfangreiche Teile enthält, die nicht analysiert werden sollen, z. B. binäre Anlagen. Die Verwendung der Inhaltsextraktion verringert die Dokumentgröße beträchtlich, erhöht dadurch die Verarbeitungsgeschwindigkeit und vermeidet gleichzeitig Analysefehler, die auf Grund ungeeigneter Daten entstehen.
- Dokumente, deren Text mit irrelevanten Textstellen durchsetzt ist, z. B. Dokumente, mit redaktionellen Informationen zwischen den Tags `<note>`. Durch das Ignorieren dieser Informationen erzielen Sie bessere Ergebnisse bei der Analyse des Dokumentinhalts.

Die native XML-Suche und die Optionen der Inhaltsextraktion in der Zuordnung eines XML-Elements zu einem UIMA-Typen widersprechen sich, da nur der gesamte Inhalt oder nur ein angegebener Inhalt berücksichtigt werden kann. Wenn Sie die Inhaltsextraktion angeben, wird die native XML-Zuordnung ignoriert. Ohne Inhaltsextraktion können Sie die Zuordnung eines XML-Elements zu einem UIMA-Typen und die native XML-Suche verwenden.

Alle Typen und Komponenten, die Sie in Ihrer Konfigurationsdatei verwenden, müssen in der Typsystembeschreibung Ihrer benutzerdefinierten Analyseschritte beschrieben sein. Sie können einen Deskriptor für ein Typsystem in Ihrer UIMA-Umgebung erstellen, indem Sie das Eclipse-Plug-in "Component Descriptor Editor" verwenden. Dieses Plug-in ermöglicht es Ihnen, auch ohne Kenntnisse der erforderlichen XML-Syntax, eine Deskriptordatei zu erstellen.

Nachdem Sie die benutzerdefinierte Analyse fertig erstellt und getestet haben, verwenden Sie den Assistenten für die Generierung für PEAR-Dateien von UIMA, um ein Archiv zu erstellen, das die benutzerdefinierten Analysedateien einschließlich der Typsystembeschreibung enthält.

Anschließend können Sie das benutzerdefinierte Analysearchiv und Ihre Konfigurationsdateien für die Zuordnung von XML-Elementen zu UIMA-Typen in die Unternehmenssuche hochladen. Verwenden Sie hierfür die Administrationskonsole für die Unternehmenssuche.

Zugehörige Tasks

„Erstellen einer Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen" auf Seite 15
In einer Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen können Sie die gesamte Palette an Konfigurationsoptionen für die Zuordnung von XML- zu UIMA-Datentypen verwenden.

Erstellen einer Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen

In einer Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen können Sie die gesamte Palette an Konfigurationsoptionen für die Zuordnung von XML- zu UIMA-Datentypen verwenden.

Informationen zu dieser Task

Die Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen muss dem im folgenden Beispiel enthaltenen Schema entsprechen.

Der XML-Beispielbericht, ein Polizeibericht, enthält XML-Tags für die Art des Verbrechens, das Tatdatum, den Tatort, den diensthabenden Polizeibeamten, seine Polizeidienststelle, die Täterbeschreibung und eine Zusammenfassung. Danach folgt ein Abschnitt mit dem Hauptteil. Beispiel:

```
<report>
 <doc>
  <crimeType>Autodiebstahl</crimeType>
  <crimeDate>04/23/05 21:23</crimeDate>
  <crimeLocation>Hauptstraße 27, Stuttgart</crimeLocation>
  <reportingOfficer rank="Lt">Jakob
   <lastname>Meier</lastName>
  </reportingOfficer>
  <policePrecinct>14. Polizeirevier</policePrecinct>
  <suspectDescription>Männlich, dunkelhaarig, dunkle Brille,
    Bluejeans und dunkles, möglicherweise schwarzes
    Jackett</suspectDescription>
  <abstract>Ein Mercedes CLK wurde am 04/23/2005 von einem Parkplatz
   vor dem Restaurant "Blaue Lagune", Hauptstraße 27 in Stuttgart
   gestohlen. (Seriennummer: 32 2761 50871)</abstract>
  <body>Ein Mercedes CLK wurde am 04/23/2005 von einem Parkplatz
   vor dem Restaurant "Blaue Lagune", Hauptstraße 27 in Stuttgart
   gestohlen. (Seriennummer: 32 2761 50871)

Er ist schwarz und hat Breitreifen der Marke Michelin.

Augenzeugen vor dem Restaurant sahen zwei dunkel gekleidete
Männer mit hoher Geschwindigkeit mit dem Auto wegfahren. Das Fahrzeug wurde
verlassen in der Ulmenallee in Bruchsal gefunden. Der Tank war leer.
Die Sitze waren stark verschmutzt und der Rücksitz wurde beschädigt.
Aus dem Fahrzeug wurde nichts gestohlen....</body>
 </doc>
 <image>
  </-- image of the crime scene as a base64-encoded string -->
 </image>
</report>
```

Auf der Basis dieses Beispielberichts, könnte eine Konfigurationsdatei die folgende Struktur aufweisen. Das Beispiel verwendet das Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

```
<?xml version="1.0"?>
<xmlCasInitializerConfiguration
 xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml">

 <identifier>Default</identifier>
 <description>Beispielkonfiguration</description>

 <contentElements>
   <element>/report/doc</element>
 </contentElements>

 <elementToTypeMappings>
```

```
  <elementToTypeMapping>
    <element>//doc//reportingOfficer</element>
   <type>com.ibm.omnifind.types.Person</type>
    <featureValueAssignment>
      <feature>role</feature>
      <basicValue default="Reporting officer">
          </basicValue>
    </featureValueAssignment>
    <featureValueAssignment>
      <feature>gender</feature>
      <basicValue default="male"
         useAttributeValue="sex"/>
    </featureValueAssignment>
    <featureValueAssignment>
      <feature>surName</feature>
      <values concatenate="true" delimiter=""/>
        <basicValue useAttributeValue="rank"
                default="Lt"/>
        <basicValue useElementContent="lastName"/>
      </values>
    </featureValueAssignment>
  </elementToTypeMapping>
  <elementToTypeMapping>
    <element>//doc</element>
    <type>com.ibm.omnifind.types.PoliceReport</type>
    <featureValueAssignment>
      <feature>crimeDescription</feature>
      <basicValue useElementContent="abstract"
               trim="true">
          </basicValue>
    </featureValueAssignment>
  </elementToTypeMapping>
 </elementToTypeMappings>

</xmlCasInitializerConfiguration>
```

Einschränkungen

Die Konfigurationsdatei für die XML-Zuordnung besteht aus zwei Abschnitten:

Element <contentElements>
: Verwenden Sie dieses Element, wenn Sie bestimmte Inhalte extrahieren wollen. Die Beispielkonfigurationsdatei extrahiert den Inhalt im Abschnitt `<doc>` eines Dokuments und ignoriert die anderen Abschnitte des Dokuments. Im XML-Polizeibericht kann eine große, für die Textverarbeitung unbrauchbare Grafik enthalten sein. Indem Sie `<doc>` als Inhaltselement angeben und nicht `<image>`, wird die Grafik herausgefiltert bevor die Textverarbeitung beginnt.

<elementToTypeMappings>
: Verwenden Sie dieses Element, um anzugeben, welche einzelnen XML-Elemente (in einem Element `<elementToTypeMapping>` angegeben) des Dokuments welchen Komponentenstrukturen der allgemeinen Analysestruktur zugeordnet werden sollen.

  Wenn Sie die Option für die Inhaltsextraktion verwenden, müssen die XML-Elemente, die im Abschnitt `<elementToTypeMappings>` angegeben sind, in den XML-Elementen enthalten sein, die im Abschnitt `<contentElements>` angegeben sind.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Konfigurationsdatei für die Zuordnung von XML- zu UIMA-Typen zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool für die XML-Prüfung, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Konfigurationsdatei heißt `configuration.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/`.
2. Nehmen Sie Ihre Zuordnungen in ein Element `<xmlCasInitializerConfiguration xmlns="http://www.ibm.com/2005/uima/jedii_ci_xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<contentElements>` hinzu, wenn Sie bestimmte Inhalte aus Dokumentabschnitten extrahieren wollen, und ein Element `<elementToTypeMappings>`, das angibt, welche einzelnen XML-Elemente des Dokuments Sie welchen Komponentenstrukturen des allgemeinen Analysebereichs zuordnen.
4. Fügen Sie ein Element `<identifier>` und ein Element `<description>` hinzu. Die Kennung legt fest, welche Konfiguration für welches XML-Dokument verwendet wird. Die Kennung muss das Stammelement des Dokuments enthalten, zum Beispiel `doc`. Wenn die Kennung auf den Standardwert gesetzt ist, ist das Stammelement des Dokuments irrelevant und die Konfigurationszuordnung wird auf jedes XML-Dokument angewendet.
5. Wenn Sie Informationen extrahieren wollen, die nur in relevanten Teilen eines Dokuments enthalten sind, fügen Sie ein Element `<contentElements>` hinzu. Es enthält das folgende Komponentenelement:
   - Mindestens ein Element `<element>`, das den Pfad zu einem XML-Element des Dokuments enthält und die XPath-Syntax einhält, z. B. `<element>/doc/crimeType</element>`.
6. Wenn Sie angeben wollen, welche XML-Elemente des Dokuments welchen Komponentenstrukturen der allgemeinen Analysestruktur zugeordnet werden, fügen Sie ein Element `<elementToTypeMappings>` hinzu. Es enthält die folgenden Komponentenelemente:
   - Mindestens ein Element `<elementToTypeMapping>`. Dieses Element muss die folgenden verschachtelten Elemente aufweisen:
     - Ein Element `<element>`, das verwendet wird, um den Pfad eines XML-Elements anzugeben, und die XPath-Syntax befolgt: Ein vorangestellter Schrägstrich (/) bedeutet, dass ein vollständiger Pfad angegeben ist. Zum Beispiel `abstract` unter dem Stammelement `doc`. Zwei Schrägstriche (//) stehen für eine beliebige Untergruppe unter einem Pfad. So muss z. B. `birthDate` innerhalb von `reportingOfficer` enthalten sein, es können jedoch andere Elemente zwischen den beiden Elementen vorhanden sein.
     - Ein Element `<type>`, das einen Typ angibt, der in der Typsystembeschreibung definiert ist. Es muss den Typ `Annotation` aufweisen.
     - Null oder mehr Elemente `<featureValueAssignment>`.
7. In einem Element `<featureValueAssignment>` benennen Sie eine Komponente des Typs `String` im Element `<feature>`, und weisen Sie im Element `<basicValue>` einen Wert zu. Mehrere Elemente `<basicValue>` können einem Element `<values>` hinzugefügt werden.

   Das Element `<basicValue>` kann Attribute haben. Zu diesen gehören `useAttributeValue`, `useElementContent`, `default` und `trim`.

Verwenden Sie `useAttributeValue`, wenn Sie den Wert eines Attributs als Wert einer Komponente verwenden wollen. Beispiel:

```
<elementToTypeMapping>
  <element>/doc//reportingOfficer</element>
  <type>com.ibm.omnifind.types.Person</type>
  <featureValueAssignment>
    <feature>role</feature>
    <basicValue default="Reporting officer"/>
  </featureValueAssignment>
  <featureValueAssignment>
    <feature>gender</feature>
    <basicValue default="male" useAttributeValue="sex"/>
  </featureValueAssignment>
</elementToTypeMapping>
```

Dieses Beispiel führt zum folgenden Ausgabeergebnis:

- Für jeden XML-Tag <reportingOfficer>, der an beliebiger Stelle innerhalb eines XML-Tags <doc> im Dokument vorhanden ist, wird eine Komponentenstruktur des Typs com.ibm.omnifind.types.Person erstellt.
- Wenn der Tag <reportingOfficer> das Attribut sex enthält, wird die Komponente gender der neu erstellten Komponentenstruktur auf diesen Attributwert gesetzt.

Verwenden Sie das Attribut useElementContent, um Inhalt als Wert einer Komponente hinzuzufügen. Das folgende Beispiel enthält einen Konfigurationsausschnitt:

```
<elementToTypeMapping>
  <element>//doc</element>
  <type>com.ibm.omnifind.types.PoliceReport</type>
  <featureValueAssignment>
    <feature>crimeDescription</feature>
    <basicValue useElementContent="abstract" trim="true"/>
  </featureValueAssignment>
</elementToTypeMapping>
```

In diesem Ausschnitt wird der Text, auf den sich das Element <abstract> in <doc> bezieht, zum Wert der Komponentenstruktur crimeDescription. Alle vorangehenden und folgenden Leerzeichen werden entfernt.

In den folgenden Fällen kann mehr als ein Wert im Element <values> angegeben werden:

- Die Komponente, die konfiguriert wird, hat den Typ StringArray.
- Durch die Verwendung des Begrenzerattributs sind viele Zeichenfolgen zu einer Zeichenfolge verknüpft und werden deshalb einer Komponente des Typs String zugeordnet. Beispiel: Der Titel Mr. ist eine Konstante, der Vorname ist ein Attributwert und der Nachname wird von einem XML-Element angegeben:

```
<elementToTypeMapping>
  <element>//doc//reportingOfficer</element>
  <type>com.ibm.omnifind.types.Person</type>
  <featureValueAssignment>
    <feature>surName</feature>
    <values concatenate="true" delimiter=" ">
      <basicValue default="Mr."/>
      <basicValue useAttributeValue="rank"
         default="Lt."/>
      <basicValue useElementContent="lastName"/>
    </values>
  </featureValueAssignment>
</elementToTypeMapping>
```

Komponentenwerte, die Zeichenfolgen enthalten werden aus der Konfigurationsdatei so extrahiert, wie sie vorliegen. Die Werte behalten etwaige vorangehende oder folgende Leerzeichen bei. Aus Namen von Typen und Komponenten werden

Leerzeichen jedoch gelöscht. So wird z. B. <type> **com.ibm.omnifind.types.Person** </type> zu <type>**com.ibm.omnifind.types.Person**</type>.

Verwenden Sie das Element <condition>, um Bedingungen für Attribute festzulegen. Die Komponentenstruktur com.ibm.omnifind.types.Person wird z. B. nur erstellt, wenn im Dokument <suspectDescription> mit dem Attribut armed, das auf yes gesetzt ist, vorhanden ist:

```
<elementToTypeMapping>
  <element>//suspectDescription</element>
  <type>com.ibm.omnifind.types.Person</type>
  <condition attribute="armed"  value="yes"/>
</elementToTypeMapping>
```

Auf der Basis des Beispielpolizeiberichts und der für die Zuordnung definierten Konfigurationsdatei werden die folgenden Komponentenstrukturen erstellt:

com.ibm.omnifind.types.PoliceReport

- covered text: ″Autodiebstahl 04/23/05 21:23 Hauptstraße 27, Stuttgart, Jakob Meier 14. Polizeirevier Männlich, dunkelhaarig, dunkle Brille, Bluejeans und dunkles, möglicherweise schwarzes Jackett Ein Mercedes CLK wurde ... Aus dem Fahrzeug wurde nichts gestohlen.
- begin = 2
- end = 904
- knownSuspects = null
- crimeDescription = ″Ein Mercedes CLK wurde am 04/23/2005 von einem Parkplatz vor dem Restaurant ″Blaue Lagune″, Hauptstraße 27 in Stuttgart gestohlen. (Seriennummer: 32 2761 50871)″

com.ibm.omnifind.types.Person

- covered text = ″Jakob Meier″
- begin = 112
- end = 127
- role = ″Reporting officer″
- firstName = null
- surName = ″Lt Meier″
- gender = ″male″

Nachdem Sie die XML-Datei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Konfigurationsdatei für die XML-Dokumentzuordnung mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

„XML-Markup in Analyse und Suche″ auf Seite 12
Sie können Informationen in XML-Strukturen in einem Dokument direkt einer allgemeinen Analysestruktur zuordnen, ohne einen UIMA-Annotator zu schreiben.

Zugehörige Verweise

„Typsystembeschreibung″ auf Seite 9
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zu Grunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Ergebnisse der Textanalyse

Alle Ergebnisse der Textanalyse werden in der allgemeinen Analysestruktur gespeichert.

Annotatoren lesen in der Regel die allgemeine Analysestruktur und schreiben auch in diese. Die privaten Anwender der allgemeinen Analysestruktur (*CAS-Anwender*) führen die Endverarbeitung der Analyseergebnisse aus, die in der allgemeinen Analysestruktur gespeichert sind. In der Unternehmenssuche gibt es zwei Arten privater CAS-Anwender:

- Der private Anwender, der den Inhalt der allgemeinen Analysestruktur in einer Suchmaschine indiziert. Für diesen privaten Anwende ist eine Konfigurationsdatei für die Indexerstellung erforderlich, die Sie mit der benutzerdefinierten Textanalyse über die Administrationskonsole für die Unternehmenssuche auswählen.
- Der private Anwender, der eine relationale Datenbank mit bestimmten Analyseergebnissen füllt. Auch für diesen privaten Anwender ist eine Konfigurationsdatei erforderlich, die Sie mit den benutzerdefinierten Textanalyseoptionen über die Administrationskonsole für die Unternehmenssuche auswählen.

Private CAS-Anwender haben nur Lesezugriff auf die allgemeine Analysestruktur.

Erforderlichenfalls können Sie benutzerdefinierte private CAS-Anwender in die Unternehmenssuche implementieren. Informationen dazu, wie Sie einen privaten Anwender schreiben, finden Sie in der Dokumentation von UIMA. Informationen dazu, wie Sie Ihren privaten Anwender in die Unternehmenssuche hochladen und verwenden, finden Sie auf der Website von IBM UIMA developerWorks unter http://www.ibm.com/developerworks/db2/zones/db2ii/.

Zugehörige Konzepte

„Indexzuordnung für benutzerdefinierte Analyseergebnisse" auf Seite 25
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Informationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

„Datenbankzuordnung für ausgewählte Analyseergebnisse" auf Seite 33
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe in der Unternehmenssuche ausgeführt haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

Komponentenpfade

Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Komponentenpfade sind sinnvoll, wenn Sie auf eine Komponentenstruktur zugreifen wollen, die komplexe Komponenten kombiniert, z. B. Komponenten, die einen Bereichswert haben oder die auf eine andere Komponentenstruktur zeigen. Wenn Sie einen Komponentenpfad verwenden, können Sie den Wert einer Komponente einer Komponentenstruktur direkt zuweisen und diesen Wert im semantischen Suchindex oder in einer Datenbank speichern.

Betrachten Sie z. B. einen Annotator, der Autos und ihre Fabrikate angibt. Er erstellt Annotationen des Typs `car`, die das Attribut `make` aufweisen. In `make` ist

jedoch nicht der Hersteller selbst enthalten (z. B. Chevrolet), sondern eine Komponentenstruktur des Typs `Company`, die wiederum das Attribut `companyname` enthält, dessen Wert eine Zeichenfolge sein muss. Wenn Sie eine semantische Abfrage aktivieren wollen, die Autonamen und Firmennamen kombiniert, verwenden Sie den Komponentenpfad `make/companyname`, um den Wert `companyname` dem Bereich `car` zuzuordnen, der für die Annotation `car` generiert wurde. Die Abfrage "Ich suche Dokumente, in denen Autos des Herstellers Chevrolet, vorkommen" wird aktiviert, indem Sie '/car[@make="Chevrolet"]' verwenden.

Ein Komponentenpfad ist eine Folge von Komponentennamen (f1/.../fn) mit den folgenden Eigenschaften:

- Der Wert eines Komponentenpfads kann eine Zeichenfolge (`String`), eine ganze Zahl (`Integer`), eine Gleitkommazahl (`Float`) oder ein Bereich eines dieser Typen sein.
- Alle Komponenten dieses Pfads von f1 bis fn-1 müssen einen komplexen Typ aufweisen, das heißt die Typen `uima.cas.TOP`, `uima.cas.FSArray`, `uima.cas.FSList` oder einen ihrer Subtypen.
- Die letzte Komponente des Pfads, fn, kann einen komplexen Typ enthalten. Darüber hinaus kann sie einen der folgenden Typen oder einen seiner Subtypen enthalten: `uima.cas.Float`, `uima.cas.Integer`, `uima.cas.String`, `uima.cas.FloatArray`, `uima.cas.IntegerArray`, `uima.cas.StringArray`, `uima.cas.FloatList`, `uima.cas.IntegerList` oder `uima.cas.StringList`.
- Optional kann eine Komponente eingegeben werden. Der vollständig qualifizierte Name des Typs muss dem Komponentennamen vorangestellt und durch einen Doppelpunkt getrennt werden. Beispiel: `f1/com.ibm.es.SomeType:f2/.../fn` .

Sie können den Typbereich einer bestimmten Komponente eingrenzen. Nehmen Sie z. B. die Komponente `additionalInfo` des Typs `uima.cas.TOP`. Wenn Sie wissen, dass der Wert Ihrer Komponente `additionalInfo` tatsächlich den Typ `EmployeeInfo` hat, der die Komponente `salary` enthält, können Sie unter Verwendung von `additionalInfo/EmployeeInfo:salary` auf diese Komponente zugreifen. Beachten Sie, dass in diesem Beispiel der Komponentenpfad `additionalInfo/salary` zu einem Fehler führen würde, da `salary` nicht für den Typ `uima.cas.TOP` definiert wurde.

Komponenten mit Bereichs- oder Listenwerten haben die folgenden zusätzlichen Eigenschaften:

- Verwenden Sie eckige Klammern ([<nummer>]), um ein bestimmtes Element des Bereichs oder der Liste auszuwählen. Ein Bereich startet bei Null (0). Wenn Sie z. B. das erste Element im Bereich `companies` auswählen wollen, verwenden Sie `companies[0]`. Die Sondermarkierung [last] kann verwendet werden, um unabhängig von der Größe den letzten Eintrag eines Bereichs auszuwählen, z. B. companies[last].
- Verwenden Sie leere eckige Klammern ([]), um alle Elemente anzugeben. In einem Komponentenpfad sind leere eckige Klammern ([]) nur einmal zulässig. Wenn Sie z. B. einen Bereich mit Verdächtigen haben, erfasst der Komponentenpfad `knownSuspects[]/com.ibm.omnifind.types.Suspect:surName` alle Nachnamen von Verdächtigen in einen Bereich `String`.
- Wird ein Komponentenpfad, der einen Bereich zurückgibt, während der Indexierung verwendet, werden die Bereichselemente verknüpft (durch Leerzeichen getrennt) und als ein aus mehreren Begriffen bestehendes Attribut oder Feld in den Index geschrieben.

- Das nächste Element des Komponentenpfads muss eingegeben werden. Der Name des Typs ist der Typ der Elemente des Bereichs. Nehmen Sie z. B. eine Komponentenstruktur des Typs `Info`. Dieser Typ hat eine Komponente namens `companies`, deren Geltungsbereich `FSArray` ist. Die Elemente des Bereichs haben den Typ `Company`. `Company`, wiederum hat eine Komponente namens `profit`. Wenn Sie nun den Gewinn des dritten Unternehmens ermitteln wollen, geben Sie folgendes ein (verwenden Sie dabei die vollständig qualifizierten Namen der Typen): `companies[3]/Company:profit`.

Integrierte Komponenten

Integrierte Komponenten sind vordefinierte Komponentennamen mit einer speziellen Semantik. Sie können verwendet werden, um auf Informationen zuzugreifen, die nicht in der Komponentenstruktur selbst enthalten ist, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich bezieht. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

Die folgenden integrierten Komponenten können in beiden Zuordnungskonfigurationsdateien verwendet werden:

- `fsId()` gibt die ID der Komponentenstruktur zurück. Die zurückgegebene ID ist eine ganze Zahl (32 Bit). Verwenden Sie diese integrierte Komponente, um auf Teile eines Dokuments zuzugreifen, die genau mit der Abfrage übereinstimmen.
- `typeName()` gibt den Objekttyp der allgemeinen Analysestruktur als Zeichenfolge zurück. Der Typ ist der vollständig qualifizierte Name des Typs, einschließlich aller Namensbereichspräfixe, z. B. `uima.tcas.Annotation`. In einem Datenbankkontext ist `typeName()` besonders nützlich, wenn Sie Typen und Subtypen in derselben Spalte speichern und wissen wollen, welches der richtige Typ einer Annotation oder einer Komponentenstruktur ist. Im folgenden Beispiel wird der Typ `person`, wie *suspect* (Verdächtiger) oder *witness* (Zeuge), in der Spalte `role` gespeichert.

```
<explicitMappingRule  applyToSubTypes="false">
    <type>com.ibm.omnifind.types.Person</type>
   <table>sample.person</table>
   <featureMappings>
     <featureMapping>
       <feature>typeName()</feature>
       <column>role</column>
     </featureMapping>
   </featureMappings>
 </explicitMappingRule>
```

- `coveredText()` gibt den Text zurück, den das allgemeine Analyseobjekt umfasst. `coveredText()` ist nur für Annotationen und ihre Subtypen verfügbar. Verwenden Sie diese integrierte Komponente nicht für Komponentenstrukturen, die nicht unter den Typ `annotation` fallen. Im folgenden Beispiel wird der Name eines Verdächtigen in der Spalte `suspectName` gespeichert.

```
<implicitMappingRule applyToSubTypes="false">
   <type>com.ibm.omnifind.types.Suspect</type>
   <relation>sample.person</relation>
   <featureMappings>
     <featureMapping>
       <feature>coveredText()</feature>
       <column>suspectName</column>
       <length>128</length>
     </featureMapping>
   </featureMappings>
 </implicitMappingRule>
```

- `[]` gibt eine Kennung an den aktuellen Containereintrag (Bereich oder Liste) zurück. Die Komponente impliziert eine Iteration, das heißt, dass ein Eintrag in der Datenbanktabelle oder dem Index für jedes Element des Bereichs oder der Liste angelegt wird. Das folgende Beispiel stammt aus einer JDBC-Konfigurationsdatei, in der die integrierte Funktion [:index] auch zulässig ist.

```
<implicitMappingRule applyToSubTypes="false">
   <type>uima.cas.FSArray</type>
   <table>beispiel.knownSuspects</table>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
       <column>arrayId</column>
     </featureMapping>
     <featureMapping>
       <feature>[:index]</feature>
       <column>arrayIndex</column>
     </featureMapping>
     <featureMapping>
       <feature>[]/com.ibm.omnifind.types.Suspect:uniqueId()</feature>
       <column>suspectId</column>
     </featureMapping>
   </featureMappings>
 </implicitMappingRule>
```

Die folgenden integrierten Komponenten können nur in der Konfigurationsdatei für die JDBC-Zuordnung verwendet werden:

- `uniqueId()` gibt die globale eindeutige ID der Komponentenstruktur zurück. Bei der zurückgegebenen eindeutigen ID handelt es sich um eine Zeichenfolge mit fester Länge (27 Zeichen) und um eine Verkettung der Ergebnisse von `fsId()`, `docId()`, `docTimestamp()` und der Anzahl aktueller Chunks, da Dokumente in der Unternehmenssuche in mehrere Chunks mit allgemeinen Analysestrukturen geteilt werden können.

  Die zurückgegebene Zeichenfolge kann alle Buchstaben von ″a-z″ und ″A-Z″, die Zahlen von ″0-9″, das Semikolon (″;″) und den Doppelpunkt (″:″) enthalten.

  Das Ergebnis von `uniqueId()` kann als Primärschlüssel für Tabellen verwendet werden.

- `objectId()` gibt die ID der Annotation oder Komponentenstruktur zurück. `objectId()` ist ähnlich wie `uniqueId()`, enthält jedoch nicht das Ergebnis von `docTimestamp()`. Die zurückgegebene ID ist nur in einer Objektgruppe eindeutig, in der die Dokument einmal syntaktisch analysiert werden. Wenn für Sie Eindeutigkeit über alle Dokumente und Dokumentversionen hinweg erforderlich ist, müssen Sie `uniqueId()` verwenden.

  Die zurückgegebene Zeichenfolge der integrierten Komponente `objectId()` hat eine feste Länge von 16 Zeichen und kann alle Buchstaben von ″a-z″ und ″A-Z″, die Zahlen von ″0-9″, das Semikolon (″;″) und den Doppelpunkt (″:″) enthalten.

  Wenn `uniqueId()` oder `objectId()` auf leere Komponentenstrukturen verweisen, wird der in der Datenbanktabellendefinition definierte Standardwert verwendet. Es werden keine leeren Objekte eines Typs gespeichert, auf den verwiesen wird.

- `docId()` gibt die Dokument-ID zurück. Der zurückgegebene Wert ist der ganzzahlige Typ `integer` (32 Bit).

  Im folgenden Beispiel werden die integrierten Komponenten angezeigt:

```
<explicitMappingRule applyToSubTypes="true">
   <type>com.ibm.omnifind.types.PoliceReport</type>
   <table>beispiel.PoliceReport</table>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
```

```
      <column>policeReportId</column>
    </featureMapping>
    <featureMapping>
      <feature>docId()</feature>
      <column>policeReportDocId</column>
    </featureMapping>
  </featureMappings>
</explicitMappingRule>
```

- `docUri()` gibt die Dokument-URI zurück.
- `docTimestamp()` gibt die Zeit (in Millisekunden) zurück, zu der das Dokument verarbeitet wurde. Diese integrierte Komponente ist sinnvoll für die Verfolgung von Dokumentversionen, z. B. wenn Sie wissen wollen, ob es sich bei der von Ihnen verwendeten Dokumentversion um die aktuellste handelt, die vom Crawler übergeben wurde.

```
<explicitMappingRule applyToSubTypes="false">
   <type>com.ibm.omnifind.types.PoliceReport</type>
   <relation>beispiel.PoliceReport</relationcolumn>
   </StoreFeature>
   <featureMappings>
     <featureMapping>
       <feature>uniqueId()</feature>
       <column>policeReportId</column>
     </featureMapping>
     <featureMapping>
       <feature>docTimestamp()</feature>
       <column>reportVersion</column>
     </featureMapping>
   </featureMappings>
</explicitMappingRule>
```

- `parentId()` gibt die `fsId()` der Komponentenstruktur zurück, die eine Containerzuordnung enthält. `parentId()` ist nur im Kontext einer Containerzuordnung gültig.
- `uniqueParentId()` gibt die `uniqueId()` der Annotation oder Komponentenstruktur zurück, der oder die in einer Containerzuordnung enthalten ist. Auch diese integrierte Komponente ist nur im Kontext einer Containerzuordnung gültig.
- `[:index]` gibt den Index des aktuellen Containereintrags zurück (Bereich oder Liste).

Zugehörige Tasks

„Abrufen von Teilen eines Dokuments, die mit einer semantischen Suchabfrage übereinstimmen" auf Seite 43
Sie können nur die Teile eines Dokuments abrufen, die genau mit der Abfrage übereinstimmen, indem Sie die relevanten Komponentenstrukturen dem Index und der Datenbank zuordnen und den Bereich in der semantischen Suchabfrage angeben.

Filter

Filter werden verwendet, um Zuordnungsregeln in den Index- und JDBC-Konfigurationsdateien zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

Das Element `<filter>` ist optional und wird verwendet, um Zuordnungen nur auf Komponenten zu beschränken, die einen bestimmten Attributwert aufweisen. Das ist sinnvoll, wenn Sie wollen, dass ein Attribut als Schalter dafür fungiert, was indexiert oder was der Datenbank hinzugfügt werden soll. So könnten z. B. Personen und Unternehmen in einer Annotation des Typs `EntityAnnotation` erfasst werden. Ihre Komponente namens `type` wird auf `person` oder auf `organization`

gesetzt. Wenn Sie nur die Personen, jedoch nicht die Unternehmen extrahieren wollen, können Sie der Zuordnungsregel den folgenden Filter hinzufügen:

```
<filter syntax="FeatureValue">type = "person"</filter>
```

Jeder Filterausdruck hat die folgende Form:

```
<FeaturePath> <Operator> <Literal>
```

Dabei gilt Folgendes:

- `FeaturePath` ist ein Komponentenpfad in der allgemeinen Analysestruktur.
- `Operator` ist =, !=, <, <=, > oder >=. Beachten Sie, dass < (und nur <) wie folgt ausgedrückt werden muss: `<`.
- `Literal` ist eine ganze Zahl, eine Gleitkommazahl (Exponentensyntax wird nicht unterstützt) oder ein in Anführungszeichen eingeschlossenes Zeichenfolgeliteral. Eingebettete Anführungszeichen und umgekehrte Schrägstriche werden mit einem umgekehrten Schrägstrich als Escapezeichen verwendet.

`<FeaturePath>`, `<Operator>` und `<Literal>` müssen mit einem Leerzeichen getrennt sein.

Die folgenden Beispiele enthalten gültige Filter:

- `<filter syntax="FeatureValue"> foo = "Hallo Welt" </filter>`

  Die Komponente `foo` enthält die Zeichenfolge `Hallo Welt`.
- `<filter syntax="FeatureValue"> foo < 42 </filter>`

  Die Komponente `foo` enthält den ganzzahligen Wert 42.
- `<filter syntax="FeatureValue"> make/company = "Chevrolet" </filter>`

  Der Komponentenpfad `make/company`, in dem die Komponente `make` eine Komponentenstruktur mit der Komponente `company` enthält, weist den Wert `Chevrolet` auf.
- `<filter syntax="FeatureValue"> bar7 >= 0.5 </filter>`

  Die Komponente `bar7` enthält den Gleitkommawert 0,5.

Indexzuordnung für benutzerdefinierte Analyseergebnisse

Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Informationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

Indem Sie Ihre Analyseergebnisse Feldern, Textbereichen und Attributen im Index für die Unternehmenssuche zuordnen, können Sie diese Informationen in Abfragen verwenden. Eine Kombination aus benutzerdefinierter Analyse und Unternehmenssuche, die in der Lage ist, sowohl Wörter als auch Textauszüge zu indexieren, ermöglicht die semantische Suche.

Indem Sie eine Konfigurationsdatei für die Indexerstellung verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen.

Sie können verschiedene Stile verwenden, um Komponentenstrukturen in der allgemeinen Analysestruktur dem Index für die Unternehmenssuche zuzuordnen.

Annotation
Wenn Sie Komponentenstukturen in der allgemeinen Analysestruktur unter Verwendung des Annotationsstils indexieren, werden alle Annotationen des angegebenen Typs im Index als durchsuchbare Bereiche gespeichert.

Wenn z. B. eine Komponentenstruktur, die einen bestimmten Textbereich umfasst, den Typ `person` aufweist und unter Verwendung des Annotationsstils indexiert wurde, sind die folgenden Abfragen möglich:

Tabelle 1. Beispiele für Abfragen

Gewünschte Information	Mögliche Abfrage
Alle Dokumente, die mindestens einen Personennamen enthalten	<person/>
Alle Dokumente, in denen in der Annotation zu einer Person "boss" vorkommt	<person>boss</person>
Alle Dokumente, in denen "Lang" im gleichen Satz wie einer meiner Konkurrenten genannt wird	<sentence><person>Lang</person> <competitor/></sentence>

Attribute von Komponentenstrukturen werden auch als Teil des Bereichs indexiert. Betrachten Sie z. B. einen Annotator, der Autos aufspürt und die Automarke als Komponente `make` der Annotation `car` speichert. Damit können Sie den folgenden Abfragetyp aktivieren: "Alle Dokumente, in denen Autos der Marke Chevrolet erwähnt werden".

Field Verwenden Sie diesen Stil, wenn Sie den Inhalt von Komponentenstrukturen während der Suche zugänglich machen wollen, indem Sie die feldspezifische Suchfunktionalität der Unternehmenssuche verwenden. Auf diese Weise kann der Inhalt einer Komponentenstruktur in den Suchergebnissen angezeigt oder in der parametrischen Suche verwendet werden.

Wenn Sie z. B. die Dosierungen von Medikamenten einem parametrischen Feld zuordnen, können Sie die folgende Abfrage verwenden: "Alle Dokumente, in denen ein bestimmtes Medikament erwähnt wird, das in einer Dosierung über 100 Milligramm eingenommen wurde."

Breaking
Verwenden Sie diesen Stil, wenn eine bestimmte Komponentenstruktur als deutlicher Begrenzer interpretiert werden soll, z. B. Abschnitte oder Absätze. Die Unternehmenssuche erkennt Sätze und Absätze standardmäßig. Verwenden Sie diesen Stil nur, wenn Ihre benutzerdefinierte Analyse zusätzliche strukturelle Elemente in einem Dokument erkennt, das Sie anders interpretieren wollen.

Analyseergebnisse können auch verwendet werden, um die Rangfolge der Dokumente in der Unternehmenssuche zu beeinflussen, selbst bei einfachen Schlüsselwortabfragen. Sie gehen hierzu in zwei Schritten vor:

1. Ordnen Sie Komponentenstrukturen durchsuchbaren Bereichen oder Feldern zu, indem Sie den Zuordnungsstil **Annotation** oder **Field** verwenden.
2. Definieren Sie eine Boostklasse in der Administrationskonsole für die Unternehmenssuche, und ordnen Sie dieser Boostklasse den Bereichs- oder Feldnamen zu.

Wenn ein Benutzer einen Suchbegriff eingibt, der in dieser Komponentenstruktur enthalten ist, wird das Dokument höher eingestuft. Betrachten Sie z. B. einen Annotator, der Personen- und Firmennamen aufspürt. Wenn Sie diese

Komponentenstrukturen Bereichen (wie ″Person″ und ″Unternehmen″) zuordnen, und anschließend diese Bereiche Boostklassen zuordnen, werden Dokumente mit den Suchergebnissen ″Lücke″ höher eingestuft, wenn es sich dabei um das Unternehmen ″Lücke″ handelt, als wenn lediglich der Begriff ″Lücke″ erwähnt wird.

Nachdem Sie die Konfigurationsdatei für die Indexerstellung geschrieben haben, können Sie diese in die Unternehmenssuche hochladen. Verwenden Sie dafür die Administrationskonsole.

Zugehörige Tasks

„Erstellen der Konfigurationsdatei für die Indexerstellung”
Indem Sie eine Konfigurationsdatei für die Indexerstellung verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

Erstellen der Konfigurationsdatei für die Indexerstellung

Indem Sie eine Konfigurationsdatei für die Indexerstellung verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

Informationen zu dieser Task

Die Konfigurationsdatei für die Indexerstellung muss dem im folgenden Beispiel dargestellten Schema entsprechen. Die Beispielkonfigurationsdatei basiert auf dem Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

```
<?xml version="1.0" encoding="UTF-8"?>
<indexBuildSpecification
xmlns="http://www.ibm.com/of/822/consumer/index/xml">
 <skipCondition>
   <type>com.ibm.uima.tt.DocumentAnnotation</type>
   <filter syntax="FeatureValue">toBeprocessed = 0</filter>
 </skipCondition>

 <indexBuildItem>
   <name>com.ibm.omnifind.types.Person</name>
   <indexRule>
     <style name="Annotation">
       <attributeMappings>
         <mapping>
           <feature>role</feature>
           <indexName>role</indexName>
         </mapping>
         <mapping>
           <feature>title</feature>
           <indexName>title</indexName>
         </mapping>
         <mapping>
           <feature>gender</feature>
           <indexName>gender</indexName>
         </mapping>
       </attributeMappings>
     </style>
   </indexRule>
 </indexBuildItem>
 <indexBuildItem>
   <name>com.ibm.omnifind.types.Suspect</name>
   <indexRule>
     <style name="Annotation"/>
     <style name="Field">
       <attribute name="parametric" value="false"/>
       <attribute name="fieldSearchable"
         value="true"/>
```

```
      <attribute name="returnable" value="true"/>
    </style>
  </indexRule>
 </indexBuildItem>
 <indexBuildItem>
  <name>com.ibm.omnifind.types.City</name>
  <indexRule>
    <style name="Annotation">
      <attributeMappings>
        <mapping>
          <feature>cityDistrict</feature>
          <indexName>district</indexName>
        </mapping>
      </attributeMappings>
    </style>
  </indexRule>
</indexBuildItem>
<indexBuildItem>
  <name>com.ibm.omnifind.types.Date</name>
  <indexRule>
    <style name="Field">
      <attribute name="fixedName" value="Date"/>
      <attribute name="fieldSearchable"
          value="true"/>
      <attribute name="returnable" value="true"/>
    </style>
    <style name="Field">
      <attribute name="fixedName" value="hour"/>
      <attribute name="valueFeature"  value="hour"/>
      <attribute name="parametric" value="true"/>
    </style>
  </indexRule>
  <filter syntax="FeatureValue">year="2005"</filter>
 </indexBuildItem>
 <indexBuildItem>
  <name>com.ibm.omnifind.types.PoliceReport</name>
  <indexRule>
    <style name="Annotation">
      <attribute name="fixedName"
         value="PoliceReport"/>
      <attributeMappings>
        <mapping>
          <feature>crimeDescription</feature>
          <indexName>crimeDescription</indexName>
        </mapping>
        <mapping>
          <feature>time/coveredText()</feature>
          <indexName>time</indexName>
        </mapping>
        <mapping>
          <feature>date/englDate</feature>
          <indexName>date</indexName>
        </mapping>
        <mapping>
          <feature>location/coveredText()</feature>
          <indexName>location</indexName>
        </mapping>
        <mapping>
          <feature>knownSuspects[]/com.ibm.omnifind.types.Suspect:surName</feature>
          <indexName>suspectsLastNames</indexName>
        </mapping>
      </attributeMappings>
    </style>
  </indexRule>
 </indexBuildItem>
</indexBuildSpecification>
```

Einschränkungen

Die Konfigurationsdatei für die Indexzuordnung muss alle Analyseergebnisse enthalten, die Sie in Abfragen durchsuchen wollen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Konfigurationsdatei für die Indexzuordnung zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Konfigurationsdatei heißt `CasToIndexMapping.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/`.
2. Nehmen Sie Ihre Zuordnungen in ein Element `<indexBuildSpecification xmlns="http://www.ibm.com/of/822/consumer/index/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<skipCondition>` hinzu, um das Indexieren bestimmter Dokumente zu verhindern. Verwenden Sie als Basis einen bestimmten Komponentenwert. Dieses Element ist optional. Im Beispiel werden Dokument nicht indexiert, wenn sie eine Datenstruktur des Typs `com.ibm.uima.tt.Document-Annotation` enthalten, deren Komponente `toBeProcessed` auf Null gesetzt ist.
4. Fügen Sie der Struktur im Index mindestens ein Element `<indexBuildItem>` hinzu, das die Zuordnung einer bestimmten Komponentenstruktur in der allgemeinen Analysestruktur enthält.
5. Speichern und prüfen Sie die XML-Datei.

Element `<indexBuildItem>`

Die Konfigurationsdatei für die Indexerstellungsspezifikation enthält mindestens ein Element `<indexBuildItem>`. Jedes Element beschreibt die Zuordnung einer bestimmten Komponentenstruktur in der allgemeinen Analysestruktur zu einer Struktur im Index (ein Bereich oder Feld).

Das Element `<name>` enthält den Typ der Komponentenstruktur. Es gibt zwei Möglichkeiten, einen Typ anzugeben:

- Den vollständigen Typnamen. Z. B. `com.ibm.omnifind.types.Suspect`
- Ein Platzhalterzeichen. Z. B. `com.ibm.omnifind.types.*`. Das Platzhalterzeichen kann nur am Ende der Typspezifikation hinzugefügt werden.

Verwenden Sie nur Subtypen von `uima.tcas.Annotation` als Elemente für die Indexerstellung. Wenn eine Komponentenstruktur den Typ `uima.cas.TOP` (statt `uima.tcas.Annotation`) aufweist, können Sie auf diese Komponentenstruktur zugreifen, indem Sie einen Komponentenpfad verwenden, der bei einer Annotation startet.

Wenn es sich bei Typ A um einen Subtyp von Typ B handelt (im Beispiel ist `com.ibm.omnifind.types.Suspect` ein Subtyp von `com.ibm.omnifind.types.Person`) und für beide Typen Elemente `<indexBuildItem>` Ia und Ib definiert sind, erfolgt die folgende Verarbeitung:

- Jede Indexierungsregel, die in Ib definiert ist, wird auf die Komponentenstrukturen des Typs B und die des Typs A angewendet

- Jede Indexierungsregel, die in Ia definiert ist, wird nur auf die Komponentenstrukturen des Typs A angewendet

Im Beispiel gilt das Element <indexBuildItem>, das für die Annotationen von com.ibm.omnifind.types.Person definiert ist, auch für die Annotationen von com.ibm.omnifind.types.Suspect. Für eine Annotation für einen Verdächtigen werden zwei Bereiche erstellt: einen für Person, der andere für Suspect.

Das Element <filter> ist optional und wird verwendet, um die Zuordnung von <indexBuildItem> nur auf Komponentenstrukturen zu beschränken, die einen bestimmten Attributwert aufweisen. Das ist sinnvoll, wenn Sie ein Attribut als Schalter verwenden wollen, um anzugeben, was indexiert werden soll. So könnten z. B. Personen und Unternehmen in einer Annotation des Typs EntityAnnotation erfasst werden. Ihre Komponente namens type wird auf person oder auf organization gesetzt. Wenn Sie nur die Personen, jedoch nicht die Unternehmen extrahieren wollen, können Sie den folgenden Filter hinzufügen:

```
<filter syntax="FeatureValue">type = "person"</filter>
```

Weiterhin könnten Sie Personen und Unternehmen unter unterschiedlichen Bereichsnamen indexieren, z. B. person und organization. Definieren Sie hierfür zwei Elemente <indexBuildItem> des Typs EntityAnnotation, und verwenden Sie zwei Filter für die Komponente type als Trigger für Personen oder Unternehmen.

Element <indexRule>

Jedes Element <indexBuildItem> enthält ein Element <indexRule>. Jedes Element <indexRule> enthält alle Informationen, die erforderlich sind, um dem Index eine Komponentenstruktur in der allgemeinen Analysestruktur als Feld-, Annotations- oder Unterbrechungsstil zuzuordnen. Der Stil **Annotation** bzw. **Field** unterstützt eine Reihe von Attributen. Den Stil **Term**, der von UIMA Software Development Kit für die Unternehmenssuche unterstützt wird, können Sie nicht verwenden. (Der Stil **Term** wird übersprungen.)

Für den Stil **Annotation** bzw. **Field** gibt es die folgenden Alternativen, um den Annotations- oder Feldnamen im Index anzugeben:

- Verwenden Sie fixedName, wenn jede Komponentenstrukur im Index unter demselben Namen zugänglich sein soll. Im folgenden Beispiel wird jede Komponentenstruktur des Typs Person einem Bereich ″Person″ im Index zugeordnet.

  ```
  <indexBuildItem>
     <name>com.ibm.omnifind.types.Person</name>
     <indexRule>
       <style name="Annotation">
         <attribute name="fixedName" value="Person" />
       </style>
     </indexRule>
  <indexBuildItem>
  ```

  Damit können Sie Abfragen wie ″Alle Dokumente, in denen ″Boss″ als Name einer Person enthalten ist″ aktivieren. Die Abfrage wird unter Verwendung von XML-Fragmenten wie folgt ausgedrückt: @xmlf2::'<Person>Boss</Person>'
- Verwenden Sie nameFeature, wenn in der Annotation verschiedene Entitäten gespeichert sind, auf die Sie unter Verwendung verschiedener Bereiche zugreifen wollen, abhängig vom Wert einer bestimmten Komponente der Annotation. Im folgenden Beispiel ist EntityAnnotation als Bereich person oder organization indexiert, was wiederum vom Wert der Komponente type abhängig ist. Bei der Komponente kann es sich auch um einen Komponentenpfad handeln.

```
<indexBuildItem>
   <name>com.ibm.tt.EntityAnotation</name>
   <indexRule>
     <style name="Annotation">
       <attribute name="nameFeature" value="type" />
     </style>
   </indexRule>
<indexBuildItem>
```

Damit können Sie Abfragen wie ″Alle Dokumente über das Unternehmen WHO″ (im Gegensatz zum englischen Begriff ″who″) aktivieren. Die Abfrage wird wie folgt in der eingeschränkten XPath-Syntax ausgedrückt:
`@xmlp::'/organization[ftcontains="WHO"]'`

- Wird keines der oben angegebenen Attribute verwendet, wird der Kurzname des Annotationstyps im Element `<indexBuildItem>` verwendet. Das ist der Standard. Beispiel:

```
<indexBuildItem>
   <name>com.ibm.uima.tutorial.RoomNumber</name>
   <indexRule>
     <style name="Annotation" />
     <style name="Field" />
   </indexRule>
 </indexBuildItem>
```

  Als Ergebnis des Elements `<indexBuildItem>` sind die Annotationen und Felder namens `RoomNumber` mit dem Text gefüllt, auf den sich `com.ibm.uima.tutorial.RoomNumber` bezieht.

Element <style name="Annotation" />

Der Wert `Annotation` im Element `<style>` gibt an, wie Sie in der Unternehmenssuche auf Bereichsinformationen zugreifen können. Außer, dass Sie die Attribute `fixedName` und `nameFeature` verwenden können, unterstützt dieser Stil auch das Element `<attributemappings>`. In diesem Element ist es möglich, den Wert einer Komponente einem Attribut zuzuordnen, das aus dem resultierenden Bereich im Index stammt, das Sie anschließend in einem Suchausdruck verwenden können.

Jede Zuordnung erfolgt in einem separaten Element `<mapping>`. Das Element `<feature>` enthält einen Komponentenpfad, und das Element `<indexName>` enthält den Namen des Attributs, das im Index verwendet wird, um den Wert von `<feature>` zu speichern. Beispiel:

```
        <mapping>
<feature>make/companyname</feature>
<indexName>company</indexName>
        </mapping>
```

Im Element `<mapping>` wird der Wert der Komponente im Pfad `make/companyname` direkt im Indexattribut `company` gespeichert.

Die Zuordnung von Komponentenwerten zu Indexattributen ist besonders sinnvoll, wenn während der Textanalyse ein komplexes Typsystem verwendet wird, das viele verschachtelte Komponentenstrukturen enthält. Wenn Sie das Element `<mapping>` verwenden, sind die relevanten Attribute ohne Korrelationsnamen, so dass Sie diese in Abfragen verwenden können, ohne detaillierte Kenntnisse der Struktur des ursprünglichen Typsystems zu haben.

Element <style name="Field" />

Der Wert `Field` im Element `<style>` gibt an, wie Sie in der Unternehmenssuche auf Feldinformationen zugreifen können. Außer den Attributen `fixedName` und `nameFeature` können Sie die folgenden Attribute definieren.

parametric
: Wenn dieses Attribut auf ″true″ gesetzt ist, kann unter Verwendung der parametrischen Suche nach dem Feldwert gesucht werden, z. B. #dosage:>100

fieldSearchable
: Wenn dieses Attribut auf ″true″ gesetzt ist, kann der Feldwert in einer Suche verwendet werden, z. B. make:Bayer

returnable
: Wenn dieses Attribut auf ″true″ gesetzt ist, werden das Feld und seine Werte in den Suchergebnissen zurückgegeben

Bei Feldinformationen kann immer der Inhalt durchsucht werden, das heißt, Feldinformationen sind für die normale Schlüsselwortsuche zugänglich.

Das optionale Attribut `valueFeature` definiert, welcher Komponentenwert als Feldwert verwendet wird. Wenn es sich bei der Komponentenstruktur um eine Annotation handelt und kein Attribut gesetzt ist, wird der von der Annotation umfasste Text als Feldwert verwendet. Beispiel:

```
<indexBuildItem>
  <name>com.ibm.omnifind.types.Date</name>
  <indexRule>
    <style name="Field">
      <attribute name="fixedName" value="date"/>
      <attribute name="fieldSearchable"
          value="true"/>
      <attribute name="returnable" value="true"/>
    </style>
    <style name="Field">
      <attribute name="fixedName" value="hour"/>
      <attribute name="valueFeature"  value="hour"/>
      <attribute name="parametric" value="true"/>
    </style>
  </indexRule>
  <filter syntax="FeatureValue">year="2005"</filter>
 </indexBuildItem>
```

Für `com.ibm.omnifind.types.Date` werden zwei Felder generiert. Das erste Feld `date` enthält den umfassten Text, z. B. 5:15pm. Das zweite Feld enthält den Wert des Attributs `hour`. Hier können Sie 'hour::<17' in einer Abfrage verwenden.

Element <style name="Breaking" />

Der Wert `Breaking` im Element `<style>` enthält keine weiteren Elemente.

Nachdem Sie die XML-Datei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Konfigurationsdatei für die Indexzuordnung mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

2 „Indexzuordnung für benutzerdefinierte Analyseergebnisse" auf Seite 25
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe ausgeführt haben, können Sie die Suchmaschine in der Unternehmenssuche verwenden, um einen Index zu erstellen. Verwenden Sie für den Index die Infor-

mationen, die in der allgemeinen Analysestruktur gespeichert sind, die von den benutzerdefinierten Analysealgorithmen erstellt wurde.

„Komponentenpfade" auf Seite 20
Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Zugehörige Verweise

„Filter" auf Seite 24
Filter werden verwendet, um Zuordnungsregeln in den Index- und JDBC-Konfigurationsdateien zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

„Typsystembeschreibung" auf Seite 9
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zu Grunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Datenbankzuordnung für ausgewählte Analyseergebnisse

Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe in der Unternehmenssuche ausgeführt haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

Diese Version unterstützt nur DB2 Universal Database Version 8.2.2 (com.ibm.db2.jcc.DB2Driver Version 2.3) und Oracle 10g (oracle.jdbc.driver.OracleDriver Version 1.0).

Bei DB2 Universal Database und Oracle können Sie auswählen, ob Sie die Analyseergebnisse direkt in die Datenbank einfügen wollen oder ob Sie die funktional entsprechenden datenbankspezifischen Ladedateien und das zugehörige Script generieren wollen, das die Ladebefehle ausführt.

Wenn Sie Ihre Analyseergebnisse Tabellen in einer Datenbank zuordnen, können Sie diese Informationen in weiteren Business-Intelligence-Verarbeitungsschritten verwenden oder damit direkt auf die relevanten Teile eines Dokuments zugreifen, die mit einer semantischen Suchabfrage übereinstimmen.

Eine Konfigurationsdatei für die XML-Zuordnung enthält Konfigurationsdaten für die Datenbankverbindung und beschreibt, welche benutzerdefinierten Analyseergebnisse in welchen Tabellen und Spalten gespeichert werden sollen. Die Tabellen- und Spaltennamen in Ihrer Konfigurationsdatei müssen den in der Datenbank erstellten Tabellen und Spalten entsprechen.

Nachdem Sie die Konfigurationsdatei erstellt haben, können Sie diese in die Unternehmenssuche hochladen. Verwenden Sie hierfür die Administrationskonsole.

Zugehörige Tasks

„Erstellen der Konfigurationsdatei für die XML-Zuordnung" auf Seite 34
Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie eine Konfigurationsdatei erstellen, die die Konfigurationsdaten für die Datenbankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Tabellen und Spalten gespeichert werden sollen.

Speichern von Analyseergebnissen in einer Datenbank

Damit Sie ausgewählte Analyseergebnisse in einer JDBC-fähigen Datenbank speichern können, müssen Sie eine Konfigurationsdatei für die Unternehmenssuche erstellen, und die erforderlichen JDBC-Treiberbibliotheken müssen sich in dem Pfad befinden, den Sie in der Konfigurationsdatei definiert haben.

Gehen Sie wie folgt vor, um Analyseergebnisse in einer JDBC-fähigen Datenbank zu speichern:

1. Entscheiden Sie, welche Analyseergebnisse Sie in der Datenbank speichern wollen. Erstellen Sie eine Datenbank, die die Tabellen mit allen erforderlichen Spalten der entsprechenden Datentypen enthält.

   Wichtig: Erstellen Sie Ihre eigene DB2-Datenbank, um die ausgewählten Analyseergebnisse darin zu speichern. Verwenden Sie dafür nicht die DB2-Datenbank, die bereits in der Installation der Unternehmenssuche enthalten ist.
2. Verwenden Sie einen XML-Editor zum Erstellen der Konfigurationsdatei für die Datenbankkonfigurationsdaten und die Analyseergebnisse, die Sie speichern wollen. Damit Sie ermitteln können, welche Analyseergebnisse in die Konfigurationsdatei aufgenommen werden sollen, müssen Sie wissen, welches zu Grunde liegende Typsystem von der benutzerdefinierten Analyse verwendet wird.
3. Stellen Sie die JDBC-Treiberbibliotheken in ein Verzeichnis, auf das der Indexknoten des Systems für die Unternehmenssuche zugreifen kann.
4. Verwenden Sie die Administrationskonsole für die Unternehmenssuche, um die Konfigurationsdatei mit der benutzerdefinierten Textanalyse hochzuladen und auszuwählen.

Erstellen der Konfigurationsdatei für die XML-Zuordnung

Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie eine Konfigurationsdatei erstellen, die die Konfigurationsdaten für die Daten- bankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Tabellen und Spalten gespeichert werden sollen.

Informationen zu dieser Task

Die Konfigurationsdatei für die XML-Zuordnung muss dem im folgenden Beispiel dargestellten Schema entsprechen. Das Beispiel basiert auf dem Typsystem, das für das Szenario des Polizeiberichts definiert wurde.

In diesem Beispiel werden nur die Polizeiberichte und die Städte, die in diesen Berichten angegeben werden, der Datenbank hinzugefügt. In diesem Beispiel wird auch die Verwendung integrierter Komponenten und die Zuordnung des Elements `<constant>` gezeigt.

```
<?xml version="1.0" encoding="UTF-8"?>
<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">
   <databaseConnection>
     <connectionUrl>db2://meinSystem:meinPort/meineDatenbank</connectionUrl>
     <driver type="jdbc">com.ibm.db2.jcc.DB2Driver</driver>

     <driverLibraries>
       <driverLibrary>C:\db2\db2jcc.jar</driverLibrary>
       <driverLibrary>C:\db2\db2jcc_license_cu.jar</driverLibrary>
       <driverLibrary>C:\db2\db2jcc_license_cisuz.jar</driverLibrary>
     </driverLibraries>
```

```
    <authentication>
      <username>meinBenutzer</username>
      <password>meinKennwort</password>
    </authentication>

    <loadFile>
      <loadFileDirectory>/home/cas2jdbc/load/</loadFileDirectory>
      <loadScript>/home/cas2jdbc/load/load.sh</loadScript>
    </loadFile>

  </databaseConnection>

  <jdbcMappingSpec>
    <skipCondition>
      <name>com.ibm.uima.tt.DocumentAnnotation</name>
      <filter syntax="FeatureValue">toBeProcessed=0</filter>
    </skipCondition>

    <cas2Jdbcmappings>
     <explicitMappings>
      <explicitMappingRule applyToSubtypes="false">
       <type>com.ibm.omnifind.types.PoliceReport</type>
       <table>sample.policeReport</table>
       <featureMappings>
        <featureMapping>
         <feature>uniqueId()</feature>
         <column>policeReportId</column>
        </featureMapping>
        <featureMapping>
         <feature>location/uniqueId()</feature>
         <column>crimeLocationId</column>
        </featureMapping>
       </featureMappings>
       <filter syntax="FeatureValue">location/coveredText()="Stuttgart"</filter>
      </explicitMappingRule>
     </explicitMappings>

      <implicitMappings>
    <implicitMappingRule applyToSubtypes="false">
        <type>com.ibm.omnifind.types.City</type>
         <table>sample.City</table>
        <featureMappings>
         <featureMapping>
          <feature>uniqueId()</feature>
          <column>crimeLocationId</column>
         </featureMapping>
         <featureMapping>
          <feature>coveredText()</feature>
          <column>cityName</column>
          <length>150</length>
        </featureMapping>
        <featureMapping>
          <constant>Deutschland</constant>
          <column>country</column>
        </featureMapping>
      </featureMappings>
       </implicitMappingRule>
      </implicitMappings>
     </cas2JdbcMappings>
    </jdbcMappingSpec>
</cas2JdbcConfiguration>
```

Einschränkungen

Erstellen Sie Ihre eigene DB2-Datenbank, um ausgewählte Analyseergebnisse darin zu speichern. Verwenden Sie dafür nicht die DB2-Datenbank, die bereits in der Installation der Unternehmenssuche enthalten ist.

Vorgehensweise

Gehen Sie wie folgt vor, um eine XML-Datenbankkonfigurationsdatei zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden. Das XSD-Schema für die Konfigurationsdatei heißt `CasToJDBCMapping.xsd` und befindet sich in Ihrer Installation der Unternehmenssuche unter *ES_INSTALL_ROOT*`/packages/uima/`.
2. Nehmen Sie Ihre Zuordnungen in ein Element `<cas2JdbcConfiguration xmlns="http://www.ibm.com/uima/consumer/jdbc/100/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<databaseConnection>` hinzu, das alle Konfigurationsdaten für die Datenbankverbindung enthält, und ein Element `<jdbcMappingSpec>`, das die Zuordnungsregeln für die Analyseergebnisse beschreibt, die in der Datenbank oder den Ladedateien gespeichert werden.
4. Fügen Sie dem Element `<databaseConnection>` die folgenden Komponentenelemente hinzu:
   - Obligatorisch: Ein Element `<connectionUrl>`. Dieses Element enthält die URL für die Datenbankverbindung. Je nach dem, welchen JDBC-Treiber Sie implementiert haben, haben Sie lokalen Zugriff oder Remotezugriff auf die Datenbank.
   - Obligatorisch: Ein Element `<driver>`. Dieses Element enthält den Namen der JDBC-Treiberklasse, z. B. `com.ibm.db2.jcc.DB2Driver` für DB2 oder `oracle.jdbc.driver.OracleDriver` für Oracle.
   - Obligatorisch: Ein Element `<driverLibraries>`. Dieses Element listet die Treiberbibliotheken auf. Jede Bibliothek ist in einem Element `<driverLibrary>` aufgelistet. Die Bibliotheken befinden sich in Ihrem DB2- oder Oracle-Installationsverzeichnis. Für DB2 gibt es die Bibliotheken `c:\ihr_db2-verz\db2jcc.jar`, `c:\ihr_db2-verz\db2jcc_license_cu.jar` und `c:\ihr_db2-verz\db2jcc_license_cisuz.jar`. Für Oracle muss die Bibliothek `c:\ihr_oracle-verz\classes12.zip` vorhanden sein.
   - Obligatorisch: Ein Element `<authentication>`. Dieses Element enthält den Benutzernamen und das Kennwort für die Datenbank.
   - Optional: Ein Element `<loadFile>`. Dieses Element enthält das Ladedateiverzeichnis in einem Element `<loadFileDirectory>` und den Namen des Ladescripts in einem Element `<loadScript>`. Wenn Sie kein Element `<loadFile>` angeben, werden die gesamten Daten unter Verwendung von JDBC direkt in der Datenbank gespeichert.

     Sie müssen außerdem alle Datenbankkonfigurationsparameter hinzufügen, wenn Sie datenbankspezifische Ladedateien und -scripts verwenden.
5. Fügen Sie dem Element `<jdbcMappingSpec>` die folgenden Komponentenelemente hinzu:
   - Optional: Ein Element `<skipCondition>`. Ist keine Bedingung zum Überspringen definiert, werden alle Dokumente verarbeitet.

```
<skipCondition>
  <name>com.ibm.uima.tt.DocumentAnnotation</name>
  <filter syntax="FeatureValue">toBeProcessed=0</filter>
</skipCondition>
```

In diesem Beispiel werden die Dokumente nicht berücksichtigt, die eine Annotation des Typs `com.ibm.uima.tt.DocumentAnnotation` enthalten, deren Komponente `toBeProcessed` auf Null gesetzt ist.

- Ein Element `<cas2JdbcMappings>`, das zeigt, welchen Datenbanktabellen und -spalten welche Typen und Komponenten zugeordnet sind. Das Element enthält einen Abschnitt für explizite und einen für implizite Zuordnungen.

6. Fügen Sie ein Element `<explicitMappings>` hinzu. Dieses Element ist obligatorisch. Es muss mindestens ein Element `<explicitMappingRule>` aufweisen, das die expliziten Zuordnungen definiert und kann nur für Annotationstypen und ihre Subtypen definiert werden. Ist im Abschnitt für explizite Zuordnungen eine Zuordnung definiert, werden alle Annotationen, die mit der Zuordnungsdefinition übereinstimmen, in der Datenbank gespeichert.
7. Optional: Fügen Sie ein Element `<implicitMappings>` hinzu. Dieses Element unterstützt alle Komponentenstrukturtypen. Wenn dieses Element vorhanden ist, muss es mindestens ein Element `<implicitMappingRule>` enthalten. Zuordnungen, die im Abschnitt für implizite Zuordnungen definiert sind, werden der Datenbank nur hinzugefügt, wenn die übereinstimmenden Annotationstypen in einer anderen Annotation angegeben sind, die mit einer expliziten oder einer impliziten Zuordnungsregel übereinstimmt.

   Eine implizite Zuordnung versetzt Sie in die Lage, nur die Analyseergebnisse zu speichern, die in einem bestimmten Kontext auftreten. Wenn z. B. die Zuordnung für eine Annotation des Typs `com.ibm.omnifind.types.City` implizit ist, werden nur die Städte in der Datenbank gespeichert, auf die von der Zuordnungsdefinition `com.ibm.omnifind.types.PoliceReport` im Abschnitt für explizite Zuordnungen verwiesen wird. Das heißt, es werden der Datenbank nur Städte hinzugefügt, die in Polizeiberichten erwähnt werden.

   Wenn es sich bei der Zuordnungsregel für die Annotation `City` um eine explizite Zuordnung handeln würde, würden der Datenbank alle Städte hinzugefügt. In beiden Fällen wird eine Stadt der Datenbank jedoch nur einmal hinzugefügt, auch wenn sie in mehreren Polizeiberichten angegeben wird.
8. Die Elemente `<explicitMappingRule>` und `<implicitMappingRule>` müssen das Attribut `applyToSubtypes` enthalten, das, wenn es auf `true` gesetzt ist, nicht nur die im Element `<type>` aufgelistete Komponentenstruktur speichert, sondern auch alle davon abgeleiteten Komponentenstrukturen. Fügen Sie den Elementen `<explicitMappingRule>` und `<implicitMappingRule>` die folgenden Komponentenelemente hinzu:
   - Ein Element `<type>`, das den Typ der Komponentenstruktur enthält.
   - Ein Element `<table>`, das das Datenbankschema und den Tabellennamen enthält. Die Syntax folgt der Regel `schema.tabellenname` oder nur `tabellenname`, wenn kein Schema definiert ist.
   - Ein Element `<featureMappings>` mit mindestens einem Element `<featureMapping>` oder einem Element `<contanerMapping>`.
   - Optional: Ein Element `<filter>`, das eine Bedingung enthält, die immer dann ausgewertet wird, wenn die Zuordnungsregel übereinstimmt. Wenn die Bedingung bei der Auswertung wahr ist, wird die Annotation oder Komponentenstruktur in der Datenbank gespeichert. Im Beispiel werden nur Polizeiberichte in der Datenbank gespeichert, in denen Verbrechen erfasst sind, die in Stuttgart begangen wurden.
9. Die Komponentenstruktur des Elements `<featureMapping>` hängt davon ab, ob Sie eine Komponente oder eine Konstante zuordnen.

Wenn Sie eine Komponente oder einen Komponentenpfad zuordnen, gehören die folgenden Elemente zu den Komponentenelementen:

- Ein Element `<feature>` mit dem Namen der Komponente. Die Komponente muss für die Komponentenstruktur im Element `type` definiert sein. Sie können auch ein Komponentenpfadkonstrukt oder eine der integrierten Komponenten verwenden.
- Optional: Ein Element `<length>`, das die Länge angibt, die eine Zeichenfolge in der angegebenen Datenbankspalte aufweisen darf. Längere Zeichenfolgen werden abgeschnitten.
- Ein Element `<column>` mit dem Namen der Spalte, in der der Komponentenwert gespeichert wird. Datenbankspalten, die nicht in Komponentenzuordnungen verwendet werden, verwenden einen in der Datenbank konfigurierten Standardwert (normalerweise Null).

  Stellen Sie sicher, dass der Wert des Komponentenelements in einer Spalte des entsprechenden Typs gespeichert wird. Der folgenden Tabelle können Sie entnehmen, welche UIMA-Typen mit welchen Datenbanktypen übereinstimmen.

Tabelle 2. Zuordnung von UIMA-Typen zu den entsprechenden Datenbanktypen

UIMA-Typ oder integrierte Komponente	**Empfohlener DB2-Datentyp**	**Empfohlener Oracle-Datentyp**
Float	REAL	FLOAT
String	VARCHAR	VARCHAR2
Integer	INTEGER	INTEGER
uniqueId(), uniqueParentId()	CHAR(27)	CHAR(27)
objectId(), parentId()	CHAR(16)	CHAR(16)
docTimestamp()	BIGINT	LONG

Eine Konstante hat die folgenden Komponentenelemente für die Komponentenzuordnung:

- Ein Element `<constant>`, das den Wert einer Konstanten enthält.
- Ein Element `<column>` mit dem Namen der Spalte, der der Wert der Konstanten hinzugefügt wird.

10. Das Element `<containerMapping>` enthält die Zuordnung für eine Containertypkomponente (Bereich oder Liste). Dieses Element darf nur für Containertypen verwendet werden. Es hat die folgenden Komponentenelemente:
    - Ein Element `<feature>` mit dem Namen der Komponente. Sie können auch ein Komponentenpfadkonstrukt oder eine der integrierten Komponenten verwenden.
    - Ein Element `<table>`, das das Datenbankschema und den Tabellennamen enthält. Die Syntax beachtet die Regel `schema.tabellenname` oder nur `tabellenname`, wenn kein Schema definiert ist.
    - Mindestens ein Element `<featureMapping>`, das die Namen der Komponentenstrukturen und Spalten enthält, denen die Komponenten hinzugefügt werden.
11. Speichern und prüfen Sie die XML-Datei mit dem bereitgestellten Schema.

Nachdem Sie die XML-Datei erstellt haben, müssen Sie diese in die Unternehmenssuche hochladen und die Konfigurationsdatei für die Datenbankzuordnung mit Ihren anderen benutzerdefinierten Analyseauswahlen in der Administrationskonsole für die Unternehmenssuche auswählen.

Zugehörige Konzepte

„Datenbankzuordnung für ausgewählte Analyseergebnisse" auf Seite 33
Nachdem Sie Ihre benutzerdefinierte Analyse für eine Dokumentobjektgruppe in der Unternehmenssuche ausgeführt haben, können Sie ausgewählte Textanalyseergebnisse in einer JDBC-fähigen Datenbank speichern.

„Komponentenpfade" auf Seite 20
Ein Komponentenpfad stellt eine Möglichkeit bereit, auf Komponentenwerte in allgemeinen Analysestrukturen zuzugreifen, ähnlich wie XPath-Anweisungen, die verwendet werden, um auf XML-Elemente in einem XML-Dokument zuzugreifen.

Zugehörige Verweise

„Filter" auf Seite 24
Filter werden verwendet, um Zuordnungsregeln in den Index- und JDBC-Konfigurationsdateien zu beschränken. Nur wenn der Filter wahr ist, werden die Analyseergebnisse dem Index oder einer JDBC-Tabelle hinzugefügt.

„Integrierte Komponenten" auf Seite 22
Integrierte Komponenten sind vordefinierte Komponentennamen mit einer speziellen Semantik. Sie können verwendet werden, um auf Informationen zuzugreifen, die nicht in der Komponentenstruktur selbst enthalten ist, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich bezieht. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

„Typsystembeschreibung" auf Seite 9
Die Typsystembeschreibung enthält die Komponentenstrukturen (die zu Grunde liegenden Strukturen, die die Analyseergebnisse darstellen), die in der benutzerdefinierten Analyse verwendet werden.

Zuordnung von Containertypen

Ein Containertyp gehört zu den integrierten Bereichs- oder Listentypen in der allgemeinen Analysestruktur. Die Zuordnung von Containertypen bietet eine Möglichkeit, einer relationalen Datenbank Bereichs- oder Listenwerte zuzuordnen.

Es gibt zwei Methoden für die Handhabung von Containertypen in der Konfigurationsdatei. Eine Methode verwendet die definierten integrierten Komponentenkonstrukte und eine generische Verknüpfungstabelle, die Bereiche oder Listen mit Werten einer Komponentenzuordnungsregel enthält. Da verschiedene Bereiche oder Listen in derselben Verknüpfungstabelle gespeichert werden, sagt die Tabelle nichts über die Relation der gespeicherten Informationen aus.

Bei der zweiten Methode wird die verwendete Definition der Verknüpfungstabelle mit einem Element `<containerMapping>` definiert und gibt die Relation zwischen den angegebenen Informationen an, nach denen Sie suchen.

So wie das folgende Beispiel könnte die Zuordnung für eine generische Verknüpfungstabelle aussehen. Es gibt eine Relation n:m zwischen Polizeiberichten und Verdächtigen, das heißt, ein Verdächtiger kann in mehreren Polizeiberichten angegeben sein, und ein Polizeibericht kann mehrere Verdächtige enthalten.

Die im Beispiel angegebene generische Tabelle `sample.fsarray` ist die Verknüpfungstabelle zwischen Polizeiberichten und Verdächtigen. Wenn ein anderer Zuordnungstyp außer `com.ibm.omnifind.types.PoliceReport` vorhanden ist, der eine Komponente des Typs `com.ibm.omnifind.types.FSArray` aufweist, wird dieser ebenfalls dieser Tabelle zugeordnet. Es ist immer noch möglich, die Tabelle nach der Relation zwischen einem Polizeibericht und einem Verdächtigen ordnungsge-

mäß abzufragen, es ist jedoch nicht möglich, durch reines Betrachten der Tabelle, den Schluss zu ziehen, dass sie eine Relation oder eine Verknüpfung zwischen Polizeiberichten und möglichen Verdächtigen enthält.

```
<cas2Jdbcmappings>
  <explicitMappings>
    <explicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.PoliceReport</type>
      <table>sample.policeReport</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>policeReportId</column>
        </featureMapping>
        <featureMapping>
          <feature>knownSuspects/uniqueId()</feature>
          <column>suspectArrayId</column>
        </featureMapping>
        <featureMapping>
          <feature>location/cityName</feature>
          <column>city</column>
        </featureMapping>
      </featureMappings>
    </explicitMappingRule>
  </explicitMappings>

  <implicitMappings>
    <implicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.Suspect</type>
      <table>sample.suspect</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>suspectID</column>
        </featureMapping>
        <featureMapping>
          <feature>surName</feature>
          <column>lastName</column>
        </featureMapping>
        <featureMapping>
          <feature>description</feature>
          <column>description</column>
        </featureMapping>
    </implicitMappingRule>
    <implicitMappingRule applyToSubtypes="false">
      <type>uima.cas.FSArray</type>
      <table>sample.fsarray</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>arrayId</column>
        </featureMapping>
        <featureMapping>
          <feature>[:index]</feature>
          <column>arrayIndex</column>
        </featureMapping>
        <featureMapping>
          <feature>[]/uniqueId()</feature>
          <column>suspectId</column>
        </featureMapping>
      </featureMappings>
    </implicitMappingRule>
  </implicitMappings>

</cas2JdbcMappings>
```

Im Folgenden werden die Datenbanktabellen auf der Basis der oben angegebenen generischen Zuordnungsregeln angezeigt.

Tabelle 3. Tabelle 'sample.policeReport'

policeReportId	**suspectArrayId**	**city**
aaa...1	**bbb...1**	Sindelfingen
aaa...2	**bbb...2**	Leonberg

Tabelle 4. Tabelle 'sample.fsarray'

arrayId	**arrayIndex**	**suspectId**
bbb...1	1	*ccc...1*
bbb...1	2	*ccc...2*
bbb...2	1	*ccc...3*

Tabelle 5. Tabelle 'sample.suspect'

suspectID	**lastname**	**description**
ccc...1	Braun	Dunkelhäutig
ccc...2	Schmidt	Brillenträger
...	...	...

Das Beispiel zeigt die Zuordnung für Komponentenstrukturbereiche. Sie können diesen Zuordnungstyp auch auf `StringArray`, `IntegerArray` und `FloatArray` anwenden. Wenn Sie für diese Bereiche mit einfachen Werten Zuordnungsregeln angeben, ersetzen Sie `[]/uniqueId()` mit `[]`.

Dieselbe Methode für generische Tabellen kann ebenfalls für Komponentenstrukturlisten verwendet werden, sowie für Listen mit einfachen Typen (`StringList`, `IntegerList` und `FloatList`).

Eine einfachere Möglichkeit, Relationen zu handhaben, besteht darin, ein Element für die explizite Containerzuordnung zu verwenden, das die Iteration für die in den Bereichen oder Listen enthaltenen Elementen definiert.

Im folgenden Beispiel wird gezeigt, wie eine Zuordnung aussieht, in der eine explizite Verknüpfungstabelle angegeben wird. Auch hier gibt es wieder die Relation n:m zwischen Polizeiberichten und Verdächtigen. Jedoch ist in diesem Fall die Tabelle `sample.reports_suspects` die Verknüpfungstabelle zwischen Polizeiberichten und Verdächtigen.

Bei dieser Methode brauchen Sie keine Bereichs-IDs oder Zuordnungen für Kopfsatz- und Nachsatzeinträge für Listentypen zu berücksichtigen. Die Verknüpfungstabelle enthält eine explizite Relation.

```
<cas2Jdbcmappings>
  <explicitMappings>
    <explicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.PoliceReport</type>
      <table>sample.policeReport</table>
      <featureMappings>
        <featureMapping>
          <feature>uniqueId()</feature>
          <column>policeReportID</column>
        </featureMapping>
```

```
        <featureMapping>
          <feature>location/cityName</feature>
          <column>city</column>
        </featureMapping>
        <featureMapping>
          <feature>knownSuspects</feature>
          <containerMapping>
            <table>sample.reports_suspects</table>
            <featureMapping>
              <feature>com.ibm.omnifind.types.PoliceReport
                /objectId()</feature>
              <column>policeReportId</column>
            </featureMapping>
            <featuremapping>
              <feature>knownSuspects/[]/objectId()</feature>
              <column>suspectId</column>
            </featureMapping>
          </containerMapping>
        </featureMapping>
      </featureMappings>
    </explicitMappingRule>
  </explicitMappings>

  <implicitMappings>
    <implicitMappingRule applyToSubtypes="false">
      <type>com.ibm.omnifind.types.Suspect</type>
      <table>sample.suspect</table>
      <featureMappings>
        <featureMapping>
          <feature>objectId()</feature>
          <column>suspectID</column>
        </featureMapping>
        <featureMapping>
          <feature>surName</feature>
          <column>lastName</column>
        </featureMapping>
        <featureMapping>
          <feature>description</feature>
          <column>description</column>
        </featureMapping>
      </featureMappings>
    </implicitMappingRule>
  </implicitMappings>

</cas2JdbcMappings>
```

Ein Element <containerMapping> wird verwendet, um die Iteration für Elemente zu definieren, die in dem Bereich enthalten sind. Im Beispiel enthält die Verknüpfungstabelle sample.reports_suspects eine Verknüpfung zu den Spalten policeReportId und suspectId. Die Elemente <containerMapping> dürfen nicht verschachtelt werden.

Im Folgenden werden die Datenbanktabellen auf der Basis der expliziten Zuordnungsregeln der Verknüpfungstabelle angezeigt.

Tabelle 6. Tabelle 'sample.policeReport'

policeReportId	city
aaa...1	Sindelfingen
aaa...2	Leonberg

Tabelle 7. Tabelle 'sample.reports_suspect'

policeReportId	**suspectId**
bbb...1	*ccc...1*
bbb...2	*ccc...2*
...	...

Tabelle 8. Tabelle 'sample.suspect'

suspectID	**lastname**	**description**
ccc...1	Braun	Dunkelhäutig
ccc...2	Schmidt	Brillenträger
...	...	...

Zugehörige Verweise

„Integrierte Komponenten" auf Seite 22
Integrierte Komponenten sind vordefinierte Komponentennamen mit einer speziellen Semantik. Sie können verwendet werden, um auf Informationen zuzugreifen, die nicht in der Komponentenstruktur selbst enthalten ist, z. B. den Typ der Komponentenstruktur oder den Annotationstext, auf den sie sich bezieht. Sie können in einem Komponentenpfad als letztes oder einziges Element verwendet werden.

Abrufen von Teilen eines Dokuments, die mit einer semantischen Suchabfrage übereinstimmen

Sie können nur die Teile eines Dokuments abrufen, die genau mit der Abfrage übereinstimmen, indem Sie die relevanten Komponentenstrukturen dem Index und der Datenbank zuordnen und den Bereich in der semantischen Suchabfrage angeben.

Wenn Sie auf alle Instanzen eines bestimmten Annotationstyps in den Suchergebnissen zugreifen wollen, z. B. um alle Personen abzurufen, fügen Sie dem Annotationstyp eine Zuordnung des Stils **Field** hinzu, und markieren Sie diesen in der Indexkonfigurationsdatei als zurückzugebend (`returnable`). Beispiel:

```
<indexBuildItem>
 <name>com.ibm.omnifind.types.Person</name>
 <indexRule>
  <style name="Annotation"/>
  <style name="Field">
   <attribute name="returnable" value="true"/>
  </style>
 </indexRule>
</indexBuildItem>
```

In diesem Beispiel werden die Annotationen des Typs `com.ibm.omnifind.types.Person` im Index für die Unternehmenssuche dem Bereich `Person` zugeordnet, wo während der semantischen Suche auf sie zugegriffen werden kann. Darüber hinaus wird der Text, auf den sich die Annotationen beziehen, z. B. der vollständige Name der Person, als zurückzugebendes Feld gespeichert. Wenn Sie diese Annotationswerte abrufen wollen, führen Sie den Aufruf `getFields("Person")` für jedes Ergebnisobjekt aus, das von der Suchabfrage (Schlüsselwort- oder semantische Suche) zurückgegeben wird. Bei dieser Methode wird ein Bereich `String` mit den Annotationswerten zurückgegeben, in diesem Fall mit den Personennamen.

Diese Methode gibt jedoch alle Instanzen eines angegebenen Annotationstyps zurück und ist deshalb nicht geeignet, wenn Sie Ihre Ergebnisverarbeitung auf Dokumente begrenzen wollen, die genau mit der Abfrage übereinstimmen. So können in einem Dokument z. B. fünf Personen erwähnt werden. Der Benutzer gibt jedoch in der semantischen Suchabfrage '<sentence><person/>IBM</sentence> an, dass er nur an der Person interessiert ist, die im selben Satz erwähnt wird, in dem auch der Begriff IBM erwähnt wird. An den anderen Personen ist der Benutzer nicht interessiert.

Gehen Sie wie folgt vor, um auf Komponentenstrukturen zuzugreifen, die genau mit der Abfrage übereinstimmen und diese zu verarbeiten:

1. Ordnen Sie die relevanten Komponentenstrukturtypen dem Index für die Unternehmenssuche zu, indem Sie den Zuordnungsstil `Annotation` verwenden. Beispiel:

```
<indexBuildItem>
 <name>com.ibm.omnifind.types.Person</name>
 <indexRule>
  <style name="Annotation"/>
 </indexRule>
</indexBuildItem>
```

2. Ordnen Sie die relevanten Komponentenstrukturtypen den JDBC-Tabellen zu. Ein Teil der Zuordnung besteht darin, dass Sie zwei Spalten für die Dokument-URI und die Komponentenstruktur-ID einfügen müssen. Obwohl Sie alle Komponentenstrukturtypen einer einzigen Datenbanktabelle zuordnen können, sollten Sie jeden Typ einer anderen Tabelle zuordnen. Beispiel:

```
      <explicitMappingRule applyToSubtypes="false">
   <type>com.ibm.omnifind.types.Person</type>
       <table>sample.person</table>
       <featureMappings>
        <featureMapping>
         <feature>objectId()</feature>
         <column>primaryId</column>
        </featureMapping>
        <!-- Contains the covered text of the annotation-->
        <featureMapping>
         <feature>coveredText()</feature>
         <column>personName</column>
        </featureMapping>
        <!-- Other mapping go in here-->
        <!-- To access the relevant person annotations in the query result-->
        <featureMapping>
         <feature>docUri()</feature>
         <column>docUri</column>
        </featureMapping>
        <featureMapping>
         <feature>fsId()</feature>
         <column>annotationId</column>
        </featureMapping>
       </featureMappings>
      </explicitMappingRule>
```

3. Führen Sie für die Dokumente eine Crawlersuche, eine syntaktische Analyse und eine Indexierung aus.
4. Rufen Sie die IDs der Instanzen ab, die mit der Abfrage übereinstimmen. In Search and Index API (SIAPI) werden diese Instanzen als Zielelemente bezeichnet. Ein Zielelement gibt den zurückzugebenden Eingabebereich an. Es wird wie folgt definiert:
   - In XML-Fragmenten wird das Zielelement von einem vorangestellten Nummernzeichen (#) angegeben. Das Nummernzeichen ist nur einmal zuläs-

sig und kann an beliebiger Stelle in der XML-Fragmentabfrage stehen. Beispiel: `$xmlf2::'<sentence><#person/>IBM</sentence>'`

- In XPath ist das Zielelement standardmäßig das letzte Feld im XPath-Ausdruck.
- Verwenden Sie die folgende Methode, um auf diese Instanzen zuzugreifen: Result.getProperty("TargetElement"). Das zurückgegebene Merkmal ist eine Verkettung von Zeichenfolgen aller Vorkommen von IDs, die durch Leerzeichen getrennt sind. Jedes Vorkommen im Merkmal kann in einen ganzzahligen Wert umgesetzt werden.

5. SIAPI gibt nicht die Komponentenstrukturen selbst zurück, sondern nur die IDs ihrer Vorkommen. Diese IDs entsprechen dem Wert `fsId()`, der in der Datenbanktabelle gespeichert ist. Ihre Anwendung muss die folgenden Schritte ausführen, um diese Instanzen und ihre zugehörigen Informationen abzurufen:
   a. Abhängig vom Bereichsnamen des Zielelements die richtige Datenbanktabelle auswählen. Im Beispiel enthält die Anwendung eine Zuordnung von `person` zur Tabelle `sample.person`. Diese Informationen werden sowohl von der Konfigurationsdatei für die Indexzuordnung abgeleitet, die den Bereichsnamen enthält, als auch von der Konfigurationsdatei für die JDBC-Zuordnung, die den Tabellennamen enthält.
   b. Für jedes Ergebnisobjekt im Suchergebnis die folgenden Schritte ausführen:
      1) Die von `Result.getProperty ("TargetElement")` zurückgegebene Zeichenfolge syntaktisch analysieren, um nach allen Element-IDs zu suchen.
      2) Eine SELECT-Anweisung für die Tabelle absetzen, indem die Ergebnis-URI (verfügbar über `Result.getDocumentId()`) als Wert in der Spalte `docUri` und die Element-IDs als Wert in der Spalte `annotationId` verwendet werden. Die Spaltennamen richten sich nach Ihrer Zuordnungsdatei. Die Spaltennamen wurden dem oben angegebenen Beispiel entnommen.

      Die zurückgegebene Zeile enthält die Informationen, die für die Komponentenstruktur gespeichert wurden, z. B. der umfasste Text oder bestimmte Attribute der Komponentenstruktur, wie "Nachname" oder "Geburtsort".

Stellen Sie sicher, dass die Aktualisierungen Ihrer Datenbank mit den Aktualisierungen des Index für die Unternehmenssuche synchronisiert werden. Wenn die Datenbank veraltete Informationen enthält (z. B. wenn Sie Datenbankladedateien verwendet und die Datenbank nicht aktualisiert haben, aber den Index aktualisiert angezeigt oder reorganisiert haben), werden möglicherweise nicht alle Element-IDs in der Datenbank gefunden. Die Unternehmenssuche behält von einen Satz immer nur die letzte Dokumentversion im Index. Das heißt, die Element-IDs gelten nur für das letzte Dokument.

Wenn Sie mehrere Versionen eines Dokuments in derselben Datenbanktabelle speichern, gibt es möglicherweise mehrere Zeilen, die übereinstimmende Element-IDs aufweisen, von denen jede aus einer anderen Version des Dokuments stammt. In diesem Fall müssen Sie eine Spalte für die Dokumentversion definieren und diese füllen, indem Sie die Anwendungslogik oder integrierte Komponenten wie `docTimestamp()` verwenden. Auf diese Weise können Sie die Ergebnisse filtern, so dass nur die letzte Dokumentversion abgerufen wird.

Zugehörige Konzepte

„Begriff der semantischen Suchabfrage" auf Seite 53
Der Begriff der semantischen Suchabfrage wird als nicht transparenter Begriff übertragen.

Zugehörige Tasks

„Erstellen der Konfigurationsdatei für die Indexerstellung" auf Seite 27
Indem Sie eine Konfigurationsdatei für die Indexerstellung verwenden, können Sie festlegen, welche Analyseergebnisse in der allgemeinen Analysestruktur Sie indexieren wollen, um die Suche zu aktivieren.

„Erstellen der Konfigurationsdatei für die XML-Zuordnung" auf Seite 34
Damit Sie einer Datenbank Analyseergebnisse hinzufügen können, müssen Sie eine Konfigurationsdatei erstellen, die die Konfigurationsdaten für die Datenbankverbindung und eine Beschreibung enthält, welche benutzerdefinierten Textanalyseergebnisse in welchen Tabellen und Spalten gespeichert werden sollen.

In der Unternehmenssuche definierte Typen und Komponenten

Das in der Unternehmenssuche definierte Typsystem umfasst die Handhabung von Metadaten und eine linguistische Basisanalyse.

Eine linguistische Basisanalyse in Form von Erkennung der Sprache und die Segmentierung eines Dokuments findet immer während der Dokumentindexierung statt, unabhängig davon, ob eine benutzerdefinierte Analyse ausgewählt ist oder nicht. Während der Basisanalyse werden der allgemeinen Analysestruktur die folgenden Informationen hinzugefügt, die Sie in Ihrer benutzerdefinierten Analyse verwenden können:

- Dokumentmetadaten des Typs `com.ibm.es.tt.DocumentMetaData`.
- Annotationen für Token, Sätze und Abschnitte der Typen `uima.tt.TokenAnnotation`, `uima.tt.SentenceAnnotation` und `uima.tt.ParagraphAnnotation`. Die Annotation für Token enthält die Komponente `lemma`.

Das für die Unternehmenssuche definierte Typsystem enthält keine hoch entwickelten textanalysespezifischen Typen und Komponenten. Diese sind im Typsystem von UIMA enthalten, das Sie verwenden und erweitern können, wenn Sie Ihre benutzerdefinierten Typen und Komponenten in Ihrer UIMA-Umgebung definieren. Möglicherweise brauchen Sie das Typsystem der Unternehmenssuche nicht zu erweitern.

Das Typsystem der Unternehmenssuche ist nicht in UIMA Software Development Kit (SDK) definiert. Wenn Sie einen dieser Typen verwenden wollen, während Sie einen Annotator in UIMA erstellen, z. B. um auf Informationen zur Dokumentsicherheit oder auf den Crawlertyp oder Dokumenttyp zuzugreifen, müssen Sie die Typen erneut in der Typsystembeschreibung Ihrer Analysesteuerkomponente definieren.

Die folgenden Typen und Komponenten sind in der Unternehmenssuche definiert:

uima.tcas.Annotation
Eine Annotation umfasst die folgenden Typen:

uima.tcas.DocumentAnnotation
Die Dokumentannotation hat die folgende Komponente:

esDocumentMetaData
Enthält Dokumentmetadaten des Typs `com.ibm.es.tt.DocumentMetaData`.

com.ibm.es.tt.ContentField
Die Inhaltsfeldannotation hat die folgende Komponente:

parameters
Die Inhaltsfeldparameter haben den Typ `com.ibm.es.tt.CommonFieldParameters`.

com.ibm.es.tt.Anchor
Die Ankerannotation für Ankertext in HTML-Dokumenten. Sie hat die folgende Komponente:

uri Die Ziel-URI des Ankertexts. Der Komponentenwert hat den Typ `uima.cas.String`.

com.ibm.es.tt.MarkupTag
Die Annotationen zu Markupinformationen, z. B. eines XML-Tags. Die Markupinformationen sind in den folgenden Komponenten gespeichert:

name Der Name des Markuptags. Der Komponentenwert hat den Typ `uima.cas.String`.

depth Die Verschachtelungstiefe. Der Komponentenwert hat den Typ `uima.cas.Integer`.

attributeName
Der Name des Komponentenattributs. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

attributeValues
Eine Wertezeichenfolge für das Attribut. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

uima.CAS.TOP
Root des Typsystems. Das Element hat die folgenden Typen:

com.ibm.es.tt.DocumentMetaData
Dokumentmetadaten haben die folgenden Komponenten. Die Komponenten werden mit der Dokumentannotationskomponente `esDocumentMetaData` verknüpft.

crawlerId
Der Crawlername. Der Komponentenwert hat den Typ `uima.cas.String`.

dataSource
Einer der folgenden Datenquellentypen:

- Web (für Dokumente, die aus dem Web-Crawler stammen).
- NNTP (für Dokumente, die aus dem Newsgroup-Crawler stammen).
- DB2 (für Dokumente, die aus dem DB2-Crawler stammen).
- Notes (für Dokumente, die aus dem Notes-Crawler stammen).
- CM (für Dokumente, die aus dem Content Management-Crawler stammen).
- FS (für Dokumente, die aus dem Crawler der UNIX-Dateisystems stammen).

- WinFS (für Dokumente, die aus dem Crawler des Windows-Dateisystems stammen).
- Exchange (für Dokumente, die aus dem Exchange-Crawler stammen).
- VBR (für Dokumente, die aus dem VeniceBridge-Crawler stammen).

Der Komponentenwert hat den Typ `uima.cas.String`.

dataSourceName
Der Name des Crawlers (Datenquelle). Der Komponentenwert hat den Typ `uima.cas.String`.

docType
Einer der folgenden Dokumenttypen:

- text/html.
- application/postscript.
- application/pdf.
- application/x-mspowerpoint.
- application/msword.
- application/x-msexcel.
- application/rtf.
- application/vnd.lotus-wordpro.
- application/x-lotus-123.
- application/vnd.lotus-freelance.
- text/xml.
- text/plain.
- application/x-js-taro (Ichitaro).

Der Komponentenwert hat den Typ `uima.cas.String`.

securityTokens
Die Token für die Dokumentsicherheit. Der Komponentenwert hat den Typ `uima.cas.StringArray`.

date Das Datum des Dokuments. Der Komponentenwert hat den Typ `uima.cas.String`.

baseUri
Die Basis-URI der Seite. Der Komponentenwert hat den Typ `uima.cas.String`.

metaDataFields
Der Komponentenwert hat den Typ `uima.cas.FSArray`. Jedes Element in diesem Bereich hat den Typ `com.ibm.es.tt.MetaDataField`.

redirectUrl
Die umgeleitete URL. Der Komponentenwert hat den Typ `uima.cas.String`.

mimeType
Der MIME-Typ oder Dokumenttyp, z. B. XML. Der Komponentenwert hat den Typ `uima.cas.String`.

url Die URL des Dokuments. Der Komponentenwert hat den Typ `uima.cas.String`.

com.ibm.es.tt.CommonFieldParameters
Zu den allgemeinen Feldparametern gehören folgende Parameter:

searchable
Eine Markierung, die angibt, ob das Feld für die freie Textsuche verfügbar ist.

fieldSearchable
Eine Markierung, die angibt, ob das Feld als Feld durchsucht wird.

parametric
Eine Markierung, die die parametrische Suche angibt.

showInSearchResult
Eine Markierung, die angibt, ob Daten in Annotationen in die Suchergebnisdetails aufgenommen werden.

resolveConflict
Eine Markierung zum Auflösen von Metadatenkonflikten zwischen `MetadataPreferred`, `ContentPreferred` und `Coexist`. Der Komponentenwert hat den Typ `uima.cas.String`.

name Der Name des Felds. Sie können unter Verwendung des Feldnamens nach diesem Feld suchen. Der Komponentenwert hat den Typ `uima.cas.String`.

com.ibm.es.tt.MetaDataField
Daten in Metadatenfeldern sind nicht Teil des Dokumentinhalts, sie werden aber in der Komponente "text" gespeichert:

parameters
Parameter von Metadatenfeldern des Typs `com.ibm.es.tt.CommonFieldParameters`.

text Der Metadatentext wird in dieser Komponente gespeichert. Sie hat den Typ `uima.cas.String`.

Zugehörige Verweise

„In UIMA definierte Typen und Komponenten"
UIMA Software Development Kit definiert linguistische Basistypen und -komponenten, die möglicherweise während der Textanalyse in einem Dokument festgestellt werden.

In UIMA definierte Typen und Komponenten

UIMA Software Development Kit definiert linguistische Basistypen und -komponenten, die möglicherweise während der Textanalyse in einem Dokument festgestellt werden.

Jede Analysesteuerkomponente hat ihre eigenen Typsystembeschreibungen, die die Eingabeanforderungen und Ausgabetypen für die Annotatoren in der Analysesteuerkomponente beschreiben. Typsystembeschreibungen sind domänen- und anwendungsspezifisch.

Sie können das UIMA-Typsystem erweitern, um Ihre eigenen Typen und Komponenten einzuschließen. In der UIMA-Umgebung gibt es ein Eclipse-Plug-in, mit dessen Hilfe Sie die Deskriptoren des Typsystems für Annotatoren editieren können. Detaillierte Informationen zum Installieren und Verwenden des Plug-ins "Component Descriptor Editor" finden Sie in der Dokumentation von UIMA.

Wenn Sie Ihre Analysesteuerkomponente in der UIMA-Umgebung vollständig entwickelt und getestet haben, enthält die von Ihnen erstellte Archivierungsdatei (PEAR-Datei) neben Ihren Dateien für die Analysesteuerkomponente auch Ihre Typsystembeschreibung.

Die folgenden Typen und Komponenten sind in UIMA definiert:

uima.tcas.Annotation
Eine Annotation umfasst die folgenden Typen:

uima.tcas.DocumentAnnotation

uima.tt.TTAnnotation

uima.tcas.DocumentAnnotation
Eine Dokumentannotation enthält die folgenden Komponenten:

categories
Eine Liste mit Kategorienamen oder -bezeichnungen für das Dokument. Der Komponentenwert hat den Typ `uima.cas.FSList`.

languageCandidates
Eine Liste der Sprachreferenzen für das Dokument. Der Komponentenwert hat den Typ `uima.cas.FSList`.

id Eine Form der Dokumentidentifikation, wie z. B. eine URL. Der Komponentenwert hat den Typ `uima.cas.String`.

uima.tt.TTAnnotation
Zu `TTAnnotation` gehören die folgenden Typen:

uima.tt.DocStructureAnnotation
Strukturinformationen zum Dokument. Die Annotation zur Dokumentstruktur schließt die folgenden Typen ein:

uima.tt.SentenceAnnotation
Ein Satz, zu dem die Anfangs- und End-Interpunktion gehört. Hierzu gehört die folgende Komponente:

sentenceNumber
Die Folgenummer des Satzes im Abschnitt. Wird am Anfang eines neuen Abschnitts auf 1 zurückgesetzt. Der Komponentenwert hat den Typ `uima.cas.Integer`.

uima.tt.ParagraphAnnotation
Ein Abschnitt. Zu seinen Komponenten gehört unter anderem Folgendes:

paragraphNumber
Die Folgenummer des Abschnitts. Der Komponentenwert hat den Typ `uima.cas.Integer`.

uima.tt.LexicalAnnotation
Informationen zum Inhalt des Dokuments. Eine lexikalische Annotation umfasst die folgenden Typen:

uima.tt.CompPartAnnotation
Ein Teil eines zusammengesetzten Worts. Zusammengesetzte Wörter bestehen in vielen germanischen Sprachen aus Wörtern, die ohne trennende Leerzeichen zusammengeschrieben werden. So besteht z. B. das Wort ″Abteilungsleiter″ aus den beiden Wörtern ″Abteilung″ und ″Leiter″.

uima.tt.TokenAnnotation
Ein Token ohne umgebende Leerzeichen. Es umfasst folgende Komponenten:

lemmaEntries
Eine Liste aller möglichen Lemmata für ein angegebenes Token. Jeder Eintrag ist ein möglicher Wörterverzeichniseintrag für das Token.

lemma
Ein Lemma aus der Liste aller möglichen Lemmata für ein Token in `lemmaEntries`. Dieses Lemma wird während der Suche verwendet.

tokenNumber
Die Folgenummer des Tokens im Satz. Wird am Anfang eines neuen Satzes auf 1 zurückgesetzt. Der Komponentenwert hat den Typ `uima.cas.Integer`.

tokenProperties
Das Merkmal eines Tokens, z. B. ist das Token in Großbuchstaben oder handelt es sich um eine Zahl. Der Komponentenwert hat den Typ `uima.cas.Integer`.

stopwordToken
Ein Token, das als Stoppwort markiert ist. Der Komponentenwert hat den Typ `uima.cas.Integer`.

synonymEntries
Eine Liste mit Verweisen auf Einträge des Typs `uima.tt.Synonym`. Jeder Eintrag ist ein möglicher Synonymeintrag für das Token.

normalizedCoveredText
Die normalisierte Darstellung des Texts, auf den sich die Annotation bezieht. Der Komponentenwert hat den Typ `uima.cas.String`.

uima.CAS.TOP
Root des Typsystems. Das Element hat die folgenden Typen:

uima.tt.KeyStringEntry
Eine Zeichenfolge mit der folgenden Komponente:

key Die Zeichenfolge.

uima.tt.Lemma
Ein Wörterverzeichniseintrag mit den folgenden morphologischen Informationen:

partOfSpeech
Eine Integralcodierung der Wortart des Lemmas.

morphID
Eine Integralcodierung der morphologischen Informationen.

uima.tt.Synonym
Ein Synonymeintrag für ein angegebenes Wort des Typs `uima.tt.keyStringEntry`.

uima.tt.LanguageConfidencePair
Ein Typ mit den folgenden Komponenten, der die Sprachauswahl des Dokuments beschreibt.

uima.tt.LanguageConfidencePair

languageConfidence
Eine Indikation (ein Gleitkommawert zwischen 0 und 1), wie gut die ausgewählte Sprache zur Dokumentsprache passt.

language
Die Sprache des Dokuments (ISO-Wert). Der Wert hat den Typ `uima.cas.String`.

languageID
Die Sprach-ID. Der Wert hat den Typ `uima.cas.Integer`.

uima.tt.CategoryConfidencePair
Ein Typ mit den folgenden Komponenten, der die Kategorieauswahl des Dokuments beschreibt.

uima.tt.CategoryConfidencePair
Eine Kategorie hat die folgenden Komponenten:

categoryString
Der Name der Kategorie. Der Wert hat den Typ `uima.cas.String`.

categoryConfidence
Eine Indikation, wie gut die Kategorie zum Dokument passt. Der Wert ist eine Gleitkommazahl.

mostSpecific
Eine Markierung (des Typs `uima.cas.Integer`), die angibt, ob die Kategorie die spezifischste für das Dokument ist.

taxonomy
Der Name der Taxonomie, zu der die Kategorie gehört. Dokumente können Kategorien verschiedener Taxonomien aufweisen. Der Wert hat den Typ `uima.cas.String`.

Zugehörige Verweise

„In der Unternehmenssuche definierte Typen und Komponenten" auf Seite 46
Das in der Unternehmenssuche definierte Typsystem umfasst die Handhabung von Metadaten und eine linguistische Basisanalyse.

Semantische Suchanwendungen

Vier Typen von Dokumentinformationen sind im Index für die Unternehmenssuche gespeichert, die Sie mit Suchanwendungen abfragen können, indem Sie Search and Index API (SIAPI) verwenden.

Die vier verschiedenen Informationstypen umfassen folgende Elemente:

- Textwörter, die in einem Dokument vorkommen, z. B. der Ausdruck *Computersoftware*.
- Bereichsnamen, z. B. ein XML-Dokument, das <author>James</author> enthält, gibt den Bereich <author> zurück.

- Attributnamen, z. B. ein XML-Dokument, das <author countryOfBirth=USA>James</author> enthält, gibt das Attribut ″countryOf-Birth″ zurück.
- Attributwerte, z. B. ist USA der Wert des Attributs ″countryOfBirth″.

Die SIAPI-Abfragesprache enthält den Begriff der semantischen Suchabfrage. Der Begriff gibt ein Zweigmuster an. Ein Zweig ist ein kleiner Baum mit Blättern. Jedes Blatt stellt die vier Informationstypen (Textwörter, Bereichsnamen usw.) dar. Die internen Knoten des Baums geben an, welche Beziehung ihre Vorkommen in einem Dokument zueinander haben. Es gibt fünf interne Knotentypen, die Beziehungen angeben:

- and
- or
- not
- in_the_span_of
- attribute_in_the_span_of

Ein Dokument entspricht einem angegebenen semantischen Suchbegriff, wenn es Vorkommen dieser Blätter aufweist und die von den internen Knoten angegebenen Integritätsbedingungen (die definierten Beziehungen) erfüllt sind.

Die semantische Suchabfrage trägt dazu bei, besser geeignete Dokumente abzurufen. Sie können nun, neben der Suche unter Verwendung Boolescher Kombinationen des Wortes mit Annotationen, auch Dokumente abrufen, in denen z. B. *James* im Bereich `Autor` vorkommt oder in denen die Begriffe *IBM* und *Suche* im selben Satz enthalten sind.

Begriff der semantischen Suchabfrage

Der Begriff der semantischen Suchabfrage wird als nicht transparenter Begriff übertragen.

Es gibt zwei Syntaxformen, um einen nicht transparenten Begriff in Search and Index API (SIAPI) auszudrücken:

- XML-Fragmente
- Xpath (eingeschränkt)

Der XML-Fragmentabfragebegriff sieht wie ein gut ausbalanciertes Fragment eines XML-Dokuments aus. Ein XML-Fragmentabfragebegriff hat als Präfix das nicht transparente Begriffszeichen `@xmlf2::`, dem der in einfache Anführungszeichen eingeschlossene XML-Fragmentausdruck folgt ('...').

Die eingeschränkten XPath-Abfragebegriffe haben das Präfix `@xmlxp::`, dem die in einfache Anführungszeichen eingeschlossene XPath-Abfrage folgt ('...').

Wie ein allgemeiner Abfragebegriff in SIAPI, kann jeder Begriff einen Modifikator für seine Darstellung haben:

Pluszeichen (+)
Der Begriff muss enthalten sein.

Präfix =
Bei dem Begriff muss es sich um eine exakte Übereinstimmung handeln.

Tilde als Präfix (~)
Es dürfen auch Synonyme des Abfragebegriffs berücksichtigt werden.

Tilde als Erweiterung (~)
Es dürfen auch Wörter berücksichtigt werden, die dasselbe Lemma wie der Abfragebegriff haben.

Die folgenden Beispiele enthalten XML-Fragmentabfragen.

@xmlf2::'<City>Sindelfingen</City>'
Sucht nach Dokumenten, die den Bereich (die Annotation) `city` mit der Zeichenfolge *Sindelfingen* enthalten.

@xmlf2::'<Person gender="female">'
Sucht nach Dokumenten, in deren Annotationen eine weibliche Person vorkommt.

@xmlf2::'<Person><.or><@gender>female</@gender>
<@title>Mrs</@title><@title>Ms</@title></.or></Person>'
Sucht nach Dokumenten, in denen eine Person anhand der Geschlechtsangabe (`gender`) oder der Anrede (`title`) als Frau erkannt wird.

@xmlf2::'<Person gender="male" role="suspect"/>
<PoliceReport><@crimeDescription><.or>Raub Diebstahl</.or>-
Unfall</@crimeDescription></PoliceReport> <City>Stuttgart<.or>
<@district>Botnang</@district><@district>Feuerbach</@district></.or></City>'
Sucht nach Dokumenten, in denen männliche Personen vorkommen, die als Verdächtige (`suspect`) eingestuft wurden, sowie eine Annotation `policeReport`, in denen die Zeichenfolgen *Raub* und *Diebstahl* in `crimeDescription` enthalten sind, nicht jedoch die Zeichenfolge *Unfall*. Weiterhin muss in diesen Dokumenten die Annotation `city` mit den Stadtteilen (`district`) *Botnang* und *Feuerbach* enthalten sein.

Die entsprechenden XPath-Abfragen haben die folgende Struktur:

@xmlxp::'//City[ftcontains(Stuttgart)]'
Sucht nach Dokumenten, die den Bereich (die Annotation) `city` mit der Zeichenfolge *Stuttgart* enthalten.

@xmlxp::'//Person[@gender="female"]'
Sucht nach Dokumenten, in deren Annotationen eine weibliche Person vorkommt.

@xmlxp::'//Person[@gender="female" or @title ftcontains(Ms) or @title ftcontains-
(Mrs)]'
Sucht nach Dokumenten, in denen eine Person anhand der Geschlechtsangabe (`gender`) oder der Anrede (`title`) als Frau erkannt wird.

@xmlxp::'//Person[@gender="male" and @role="suspect"] //PoliceReport [@crime-
Description ftcontains(Raub) or @crimeDescription ftcontains(Diebstahl)] //City [
(@district="Botnang" or @district="Feuerbach") and ftcontains(Stuttgart)]'
Sucht nach Dokumenten, in denen männliche Personen vorkommen, die als Verdächtige (`suspect`) eingestuft wurden, sowie eine Annotation `policeReport`, in denen die Zeichenfolgen *Raub* und *Diebstahl* in `crimeDescription` enthalten sind. Weiterhin muss in diesen Dokumenten die Annotation `city` mit den Stadtteilen (`district`) *Botnang* und *Feuerbach* enthalten sein.

Synonymunterstützung in Suchanwendungen

Benutzer können die Suchergebnisse erweitern, indem sie nach Dokumenten suchen, die Synonyme der Abfragebegriffe enthalten.

Zu Synonymen zählen in der Regel Mehrwortbegriffe, wie z. B. Produktnamen wie *WebSphere Information Integrator OmniFind*. Mehrwortbegriffe, die im Synonymverzeichnis enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Die SIAPI-Schnittstelle für die Unternehmenssuche unterstützt verschiedene Methoden, mit denen Benutzer nach Synonymen der Abfragebegriffe suchen können:

- Die SIAPI-Abfragesyntax unterstützt die Tilde (~) als Operator für die Synonymerweiterung. Wenn der Benutzer die Tilde einem Abfragebegriff voranstellt, wird für dieses Wort eine Synonymerweiterung durchgeführt. Die Abfrage `~WAS` beispielsweise gibt Dokumente zurück, die WebSphere Application Server und andere vorhandene Synonyme für diese Abkürzung behandeln.
- Die Synonymerweiterung kann über die SIAPI-Synonymerweiterungsschnittstelle in einer Suchanwendung aktiviert werden. Abfragebegriffe können automatisch so erweitert werden, dass sie Synonyme mit einschließen, oder die Suchanwendung kann Optionen enthalten, mit denen der Benutzer angeben kann, ob Synonyme der Abfragebegriffe als Suchergebnisse zurückgegeben werden sollen.

  Bei der automatischen Synonymerweiterung wird die Synonymsuche für alle Abfragewörter und Inhaltsfelder durchgeführt. Die Suchergebnisse enthalten Dokumente, die die Abfragebegriffe oder Synonyme der Abfragebegriffe enthalten. Die Suchergebnisse zeigen auch an, welche Begriffe auf welche Synonyme erweitert wurden.

  Bei der benutzergesteuerten Variante zeigt die Suchanwendung vor der Durchführung der Abfrage dem Benutzer an, welche Synonyme für jedes Abfragewort gefunden wurden. Der Benutzer wählt dann die Begriffe aus, die in die Suche einbezogen werden sollen, oder er formuliert die Suche neu und entfernt ursprüngliche Abfragebegriffe. In diesem Szenario entscheidet der Benutzer, welche Begriffe in die Abfrage aufgenommen werden sollen: nur genaue Entsprechungen oder unterschiedliche Wortbedeutungen und -verwendungen.

Erstellen einer XML-Datei für Synonyme

Zum Erweitern von Abfragen in einer Unternehmenssuche um Synonyme der Abfragebegriffe müssen Sie in einer XML-Datei festlegen, welche Wörter als Synonyme voneinander gelten.

Informationen zu dieser Task

Die XML-Datei, die die Synonyme auflistet, muss dem im folgenden Beispiel dargestellten Schema entsprechen.

```
<?xmlversion="1.0" encoding="UTF-8"?>
<synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml">
  <synonymgroup>
    <synonym>Think Pad</synonym>
    <synonym>Notebook</synonym>
    <synonym>Notebooks</synonym>
  </synonymgroup>
```

```
  <synonymgroup>
    <synonym>WebSphere Application Server</synonym>
    <synonym>WAS</synonym>
  </synonymgroup>
</synonymgroups>
```

Einschränkungen

Wörter, die synonym sind (Elemente <synonym>), müssen Sie in einem Element <synonymgroup> zusammenfassen. Ein Synonym kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

Sie müssen alle möglichen Beugungen der Begriffe aufführen, die Sie als Synonym hinzufügen, wie z. B. die Singular- und Pluralformen der Wörter. Sie müssen jedoch nicht die normalisierten Formen eines Begriffs auflisten, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch). Wenn Sie beispielsweise das Wort "météo" als Synonym hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Synonymliste für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden.
2. Fügen Sie ein Element <synonymgroup> hinzu, und fügen Sie dann für jedes Wort, das in der Synonymgruppe als Synonym für andere Wörter behandelt werden soll, ein Element <synonym> ein.

   Achten Sie darauf, Ihre Zuordnungen in ein Element <synonymgroups xmlns="http://www.ibm.com/of/822/synonym/xml"> aufzunehmen. Der Namespace (im Attribut xmlns angegeben) muss genauso wie dargestellt angegeben werden.
3. Wiederholen Sie den vorigen Schritt, bis Sie alle Synonyme angegeben haben, die Sie für die Suche nach Dokumenten in einer Objektgruppe für die Unternehmenssuche verwenden wollen.
4. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Synonymverzeichnis umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Synonymverzeichnisses

Nach dem Erstellen oder Aktualisieren einer Synonymliste in einer XML-Datei müssen Sie die XML-Datei in ein Synonymverzeichnis umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Synonymverzeichnisses das Befehlszeilentool **essyndictbuilder**, das zusammen mit WebSphere II OmniFind Edition geliefert wird. Das Tool befindet sich im Verzeichnis *ES_INSTALL_ROOT*/bin.

Als Eingabe für das Tool dient die XML-Datei, die die Synonyme auflistet. Die Ausgabe ist ein Synonymverzeichnis. Das Wörterverzeichnis muss das Suffix .dic haben. Beispiel: c:\eigeneVerzeichnisse\produkte.dic.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Synonymverzeichnis für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von WebSphere II OmniFind Edition angegeben.
2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Synonymliste und *dic-datei* der vollständig qualifizierte Pfad zum Synonymverzeichnis.

   AIX, Linux oder Solaris: `essyndictbuilder.sh` *xml-datei dic-datei*
   Windows: `essyndictbuilder.bat` *xml-datei dic-datei*

Fügen Sie nach dem Erstellen des Synonymverzeichnisses über die Administrationskonsole für die Unternehmenssuche das Synonymverzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte `DIC`-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird, und sichern Sie die Datei regelmäßig. Sie brauchen diese XML-Datei, um Ihr Synonymverzeichnis zu aktualisieren.

Benutzerdefinierte Verzeichnisse von Stoppwörtern

Benutzer können ein unternehmensspezifisches Vokabular definieren, das aus einer Abfrage entfernt wird, um die Suchrelevanz zu erhöhen.

Es gibt zwei Arten der Stoppwortunterstützung in der Unternehmenssuche:

- Die sprachspezifische Stoppworterkennung, die alle häufig verwendeten Wörter, wie *ein* und *der* aus einer Mehrwortabfrage entfernt. Die Verzeichnisse von Stoppwörtern, die für die einzelnen Sprachen vorhanden sind, können von den Benutzern nicht geändert werden. Diese Stoppworterkennung wird für alle Abfragen automatisch ausgeführt, um die Suchrelevanz zu verbessern.
- Die benutzerdefinierte oder angepasste Stoppworterkennung, die unternehmensspezifisches Vokabular aus Abfragen entfernt. Dieses Verzeichnis von Stoppwörtern, das vom Administrator definiert wird, kann nur ein spezielles Vokabular enthalten. Das benutzerdefinierte Verzeichnis von Stoppwörtern ersetzt in der Unternehmenssuche nicht die sprachspezifischen Verzeichnisse von Stoppwörtern, die allgemeine Wörter enthalten.

Zu benutzerdefinierten Stoppwörtern zählen in der Regel Mehrwortbegriffe, wie z. B. Produktnamen wie *WebSphere Information Integrator OmniFind*. Mehrwortbegriffe, die im Verzeichnis von Stoppwörtern enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Auch die zusammengesetzten Begriffe der germanischen Sprachen werden in Abfragen richtig identifiziert. Ein zusammengesetzter Begriff ist eine Kombination aus mindestens zwei Begriffen, die wie ein einziger Begriff verwendet werden. Lexikalisierte Zusammensetzungen wie *Reisebüro* gelten nicht als zusammengesetzte Begriffe.

In einer Abfrage werden die zusammengesetzten Begriffe in die Einzelbegriffe zerlegt, aus denen sie zusammengesetzt sind. Falls einer der Einzelbegriffe eines zusammengesetzten Begriffs im Verzeichnis von Stoppwörtern enthalten ist, wird der zusammengesetzte Begriff nicht aus der Abfrage entfernt.

So gibt z. B. der Abfragebegriff `Versicherungspolice` Dokumente zurück, die die zusammengesetzten Begriffe *Lebensversicherungspolice* und *Haftpflichtversicherungspolice* enthalten. Der zweite Begriff wird ebenfalls für die Abfrage `Haftpflicht` zurückgegeben. Selbst wenn der Begriff *Police* im Verzeichnis von Stoppwörtern aufgeführt ist, wird der zusammengesetzte Abfragebegriff `Versicherungspolice` nicht aus der Abfrage entfernt.

Sie müssen das unternehmensspezifische Vokabular in einer XML-Datei auflisten, die Sie anschließend in ein Verzeichnis von Stoppwörtern konvertieren müssen, damit dieses dem System für die Unternehmenssuche hinzugefügt werden kann.

Über die Administrationskonsole für die Unternehmenssuche können Sie auswählen, welches Verzeichnis von Stoppwörtern verwendet wird. Sie können für jede Objektgruppe ein Verzeichnis von Stoppwörtern auswählen. Ein Verzeichnis von Stoppwörtern kann von mehreren Objektgruppen gemeinsam genutzt werden.

Erstellen einer XML-Datei für Stoppwörter

Damit Sie unternehmensspezifisches Vokabular aus Abfragen entfernen können, müssen Sie in einer XML-Datei angeben, welche Wörter Sie als Stoppwörter verwenden wollen.

Informationen zu dieser Task

Die XML-Datei, die die Stoppwörter auflistet, muss dem im folgenden Beispiel dargestellten Schema entsprechen.

```
<?xml version="1.0" encoding="UTF-8"?>
<stopWords xmlns="http://www.ibm.com/of/83/stopwordbuilder/xml">
   <stopWord>OmniFind Edition</stopWord>
   <stopWord>WAS<stopWord>
   <stopWord>...</stopWord>
</stopWords>
```

Einschränkungen

Ein Stoppwort kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

Sie müssen jedoch nicht die normalisierten Formen eines Begriffs auflisten, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch). Wenn Sie beispielsweise das Wort ″météo″ als Stoppwort hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Liste von Stoppwörtern für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool für die XML-Prüfung, um XML-Syntaxfehler zu vermeiden.
2. Fügen Sie für jedes Wort, das als Stoppwort behandelt werden soll, ein Element `<stopWord>` hinzu.

   Achten Sie darauf, Ihre Zuordnungen in ein Element `<stopWords xmlns=″http://www.ibm.com/of/83/stopwordbuilder/xml″>` aufzunehmen. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Wiederholen Sie diesen Schritt, bis Sie alle Stoppwörter angegeben haben, die aus Abfragen entfernt werden sollen, wenn Benutzer die Objektgruppen für die Unternehmenssuche durchsuchen.
4. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Verzeichnis von Stoppwörtern umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Verzeichnisses von Stoppwörtern

Nach dem Erstellen oder Aktualisieren einer Liste von benutzerdefinierten Stoppwörtern in einer XML-Datei müssen Sie die XML-Datei in ein Verzeichnis von Stoppwörtern umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Verzeichnisses von Stoppwörtern das Befehlszeilentool **esstopworddictbuilder**, das zusammen mit WebSphere II OmniFind Edition geliefert wird. Das Tool befindet sich im Verzeichnis *ES_INSTALL_ROOT*/bin.

Als Eingabe für das Tool dient die XML-Datei, die die Stoppwörter auflistet. Die Ausgabe ist ein Verzeichnis von Stoppwörtern. Das Wörterverzeichnis muss das Suffix .dic haben. Beispiel: c:\eigeneVerzeichnisse\produktstoppwörter.dic.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Verzeichnis von Stoppwörtern für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von WebSphere II OmniFind Edition angegeben.
2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Liste mit Stoppwörtern und *dic-datei* der vollständig qualifizierte Pfad zum Verzeichnis von Stoppwörtern.

   AIX, Linux oder Solaris: esstopworddictbuilder.sh *xml-datei dic-datei*
   Windows: esstopworddictbuilder.bat *xml-datei dic-datei*

Fügen Sie nach dem Erstellen des Verzeichnisses von Stoppwörtern über die Administrationskonsole für die Unternehmenssuche das Verzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte DIC-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird, und sichern Sie die Datei regelmäßig. Sie brauchen diese XML-Datei, um Ihr Verzeichnis von Stoppwörtern zu aktualisieren.

Benutzerdefinierte Verzeichnisse von Boostwörtern

Benutzer können bestimmte Begriffe oder Mehrwortbegriffe definieren, die den Rangordnungswert des Dokuments, das einen dieser Begriffe enthält, höher oder niedriger einstufen.

Jedem Begriff des Verzeichnisses ist ein Boostfaktor in einem Bereich zwischen -10 und +10 zugeordnet. Den Begriffen, die Sie im Ergebnisdokument als besonders wichtig erachten, ordnen Sie einen höheren Boostfaktor zu, während Sie denjenigen, die gar nicht oder nur in Kombination mit höherwertigen Boostbegriffen angezeigt werden sollen, einen niedrigeren Wert zuordnen. Die Werte -1, 0 und 1 haben keine Boostwirkung.

Wird ein Abfragebegriff, der im Verzeichnis von Boostwörtern mit einem bestimmten Boostfaktor aufgelistet ist, in einem abgerufenen Dokument angezeigt, wird der Rangordnungswert des Dokuments abhängig vom Boostwert erhöht oder erniedrigt. Der einem Begriff zugeordnete Boostwert ist relativ, da er auch von anderen Faktoren beeinflusst wird. Das heißt, wenn der Begriff X den Boostwert B1 und der Begriff Y den Boostwert B2 hat, und B1 > B2 ist, ist die Boostwirkung (X) >= Boostwirkung (Y).

Zu Boostwörtern zählen in der Regel Mehrwortbegriffe wie Produktnamen wie z. B. *WebSphere Information Integrator OmniFind*. Mehrwortbegriffe, die im Verzeichnis von Boostwörtern enthalten sind, werden in Benutzerabfragen richtig identifiziert und müssen nicht in Anführungszeichen gesetzt werden.

Auch die zusammengesetzten Begriffe der germanischen Sprachen werden in Abfragen richtig identifiziert. Ein zusammengesetzter Begriff ist eine Kombination aus mindestens zwei Begriffen, die wie ein einziger Begriff verwendet werden. Lexikalisierte Zusammensetzungen wie *Reisebüro* gelten nicht als zusammengesetzte Begriffe.

In einer Abfrage werden die zusammengesetzten Begriffe in die Einzelbegriffe zerlegt, aus denen sie zusammengesetzt sind. Wenn Boostwerte aus den Einzelbegriffen eines zusammengesetzten Begriffs bestehen, werden die abgerufenen Dokumente eingestuft. Allerdings ist der zugeordnete Wert niedriger, als bei Begriffen, die nicht Teil eines zusammengesetzten Begriffs sind. Dadurch wird der Suchbereich erweitert, was immer dann sinnvoll ist, wenn nur wenige Dokumente gefunden werden, die den vollständigen zusammengesetzten Begriff enthalten.

So gibt z. B. der Abfragebegriff `Versicherungspolice` Dokumente zurück, die die zusammengesetzten Begriffe *Lebensversicherungspolice* und *Haftpflichtversicherungspolice* enthalten. Der letztgenannte Begriff wird ebenfalls für die Abfrage `Haftpflicht` zurückgegeben. Wenn das Wort *Police* im Verzeichnis von Boostwörtern vorhanden ist, wird dem Dokument, das den zusammengesetzten Abfragebegriff `Versicherungspolice` enthält, ein Boostwert zugeordnet.

Sie müssen die Begriffe mit ihrem Boostwert in einer XML-Datei auflisten und diese anschließend in ein Verzeichnis von Boostwörtern konvertieren, das Sie dem System für die Unternehmenssuche hinzufügen können.

Über die Administrationskonsole für die Unternehmenssuche können Sie auswählen, welches Verzeichnis von Boostwörtern verwendet wird. Für jede Objektgruppe

können Sie ein Verzeichnis von Boostwörtern auswählen. Ein Verzeichnis von Boostwörtern kann von mehreren Objektgruppen gemeinsam genutzt werden.

Erstellen einer XML-Datei für Boostwörter

Damit Sie die Wertigkeit bestimmter Ergebnisdokumente erhöhen oder erniedrigen können, müssen Sie in einer XML-Datei angeben, welche Wörter die Rangordnung von Dokumenten beeinflussen können.

Informationen zu dieser Task

Die XML-Datei, die die Boostwörter auflistet, muss dem im folgenden Beispiel dargestellten Schema entsprechen.

```
<?xml version="1.0" encoding="UTF-8"?>
<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">
    <!-- group boost terms by boost value-->
   <boostTermList boost="5">
    <!-- each term can specify the synonym expansion separatly-->
     <term useVariants="true">OmniFind Edition</term>
     <term useVariants="false">Edition</term>
         <term useVariants>OmniFind</term>
   </boostTermList>
   <boostTermList boost="8">
        <term useVariants="true">WAS</term>
        <term>term9</term>
    </boostTermList>
</boostTerms>
```

Einschränkungen

Sie müssen Begriffe, die denselben Boostwert aufweisen in einem Element `<boostTermList>` gruppieren. Ein Boostwert kann mehr als einmal auftreten, z. B. wenn Sie Boostwörter alphabetisch in der XML-Datei sortieren wollen.

Ein Boostwort kann Leerzeichen enthalten, jedoch keine Interpunktionszeichen wie Kommata (,) oder senkrechte Striche (|), da diese die Abfragesyntax der Unternehmenssuche stören würden.

In der Regel haben Boostwörter Varianten, wie Akronyme oder Abkürzungen. Sie können alle Varianten im Verzeichnis von Boostwörtern aufführen. Wenn Sie jedoch beabsichtigen, auch ein Synonymverzeichnis zu verwenden, und Sie dem Synonymverzeichnis bereits Begriffe mit ihren Varianten hinzugefügt haben, ist es nicht erforderlich, diese Varianten ebenfalls dem Verzeichnis von Boostwörtern hinzuzufügen. Sie können stattdessen für die Varianten, die Sie dem Verzeichnis von Boostwörtern hinzufügen, einfach das Attribut `useVariants` auf `true` setzen. Alle im Synonymverzeichnis aufgelisteten Varianten dieses Begriffs, die in einem der abgerufenen Dokumente enthalten sind, beeinflussen die Rangfolge, die diesen Dokumenten zugewiesen wird.

Sie müssen jedoch nicht die normalisierten Formen eines Begriffs auflisten, wie z. B. das Entfernen von Akzenten und Umlauten (die Unternehmenssuche normalisiert automatisch). Wenn Sie beispielsweise das Wort ″météo″ als Boostwort hinzufügen möchten, müssen Sie nicht auch das Wort METEO aufnehmen.

Vorgehensweise

Gehen Sie wie folgt vor, um eine Liste von Boostwörtern für die Unternehmenssuche zu erstellen:

1. Erstellen Sie eine XML-Datei. Verwenden Sie einen XML-Editor oder ein XML-Authoring-Tool Ihrer Wahl, um XML-Syntaxfehler zu vermeiden.
2. Nehmen Sie ihre Zuordnungen in ein Element `<boostTerms xmlns="http://www.ibm.com/of/83/boostbuilder/xml">` auf. Der Namespace (im Attribut `xmlns` angegeben) muss genauso wie dargestellt angegeben werden.
3. Fügen Sie ein Element `<boostTermList>` hinzu, um alle Begriffe, die denselben Boostwert verwenden zu gruppieren.

   Die Boostwerte bewegen sich im Bereich zwischen -10 und 10. Beispiele: `<boostTermList boost="-5">` oder `<boostTermList boost="5">`.

   Die Wertigkeit der Dokumente, die die angegebenen Begriffe enthalten, wird anhand des angegebenen Boostwerts erhöht oder erniedrigt.
4. Fügen Sie für jeden Begriff, der den angegebenen Boostwert verwendet, ein Element `<term>` hinzu.

   Wenn Sie auch die Varianten eines Boostworts aufnehmen wollen, die in einem Synonymverzeichnis aufgelistet sind, setzen Sie das Attribut `useVariants` des Elements `<term>` auf `true`. Der Standardwert ist `false`. Es wird keine Fehlernachricht erzeugt, wenn keine Varianten im Synonymverzeichnis gefunden werden.
5. Wiederholen Sie diese Schritte, bis Sie alle Begriffe angegeben haben, die die Benutzer zum Durchsuchen der Objektgruppen für die Unternehmenssuche als Boostwörter verwenden sollen.
6. Speichen und beenden Sie die XML-Datei.

Nach dem Erstellen der XML-Datei müssen Sie sie in ein Verzeichnis von Boostwörtern umwandeln, das Sie dem System für die Unternehmenssuche hinzufügen können.

Erstellen eines Verzeichnisses von Boostwörtern

Nach dem Erstellen oder Aktualisieren einer Liste von Boostwörtern in einer XML-Datei müssen Sie die XML-Datei in ein Verzeichnis von Boostwörtern umwandeln.

Informationen zu dieser Task

Verwenden Sie für die Erstellung eines Verzeichnisses von Boostwörtern das Befehlszeilentool **esboostworddictbuilder**, das zusammen mit WebSphere II OmniFind Edition geliefert wird. Das Tool befindet sich im Verzeichnis `ES_INSTALL_ROOT/bin`.

Als Eingabe für das Tool dient die XML-Datei, die Ihre Boostwörter auflistet. Die Ausgabe ist ein Verzeichnis von Boostwörtern. Das Wörterverzeichnis muss das Suffix `.dic` haben. Beispiel: `c:\eigeneVerzeichnisse\produktboostwörter.dic`.

Standardspeicherort für beide Dateien ist das Verzeichnis, in dem das Script aufgerufen wird. Wenn bereits ein Verzeichnis mit dem gleichen Namen vorhanden ist, generiert das Script einen Fehler.

Vorgehensweise

Gehen Sie wie folgt vor, um ein Verzeichnis von Boostwörtern für die Unternehmenssuche zu erstellen:

1. Melden Sie sich am Indexserver als Administrator für die Unternehmenssuche an. Diese Benutzer-ID wurde bei der Installation von WebSphere II OmniFind Edition angegeben.

2. Geben Sie den folgenden Befehl ein. Dabei ist *xml-datei* der vollständig qualifizierte Pfad zur XML-Datei mit der Liste mit Boostwörtern und *dic-datei* der vollständig qualifizierte Pfad zum Verzeichnis von Boostwörtern. Wenn Sie gleichzeitig ein Synonymverzeichnis verwenden wollen, fügen Sie den vollständig qualifizierten Pfad zum Synonymverzeichnis nach dem Namen des Verzeichnisses von Boostwörtern hinzu. Die Angabe von Synonymverzeichnissen ist optional.

   UNIX: `esboostworddictbuilder.sh` *xml-datei dic-datei synverz-datei*
   Windows: `esboostworddictbuilder.bat` *xml-datei dic-datei synverz-datei*

Fügen Sie nach dem Erstellen des Verzeichnisses von Boostwörtern über die Administrationskonsole für die Unternehmenssuche das Verzeichnis dem System für die Unternehmenssuche hinzu, und ordnen Sie es mindestens einer Objektgruppe zu.

Nur die generierte `DIC`-Datei wird auf das System für die Unternehmenssuche hochgeladen. Stellen Sie sicher, dass die XML-Datei in einer Umgebung mit Zugriffssteuerung aufbewahrt wird und eine geeignete Backup-Strategie aktiviert ist. Sie brauchen diese XML-Datei, um Ihr Verzeichnis von Boostwörtern zu aktualisieren.

Zugehörige Tasks

„Erstellen eines Synonymverzeichnisses" auf Seite 56
Nach dem Erstellen oder Aktualisieren einer Synonymliste in einer XML-Datei müssen Sie die XML-Datei in ein Synonymverzeichnis umwandeln.

Textanalyse innerhalb der Unternehmenssuche

Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

Bei der Verarbeitung eine Dokuments ermittelt die Unternehmenssuche die Sprache dieses Dokuments und unterteilt den Eingabedatenstrom in eindeutige Einheiten oder Token.

Während einer Suche muss der Benutzer oder eine Anwendung die Abfragesprache manuell auswählen. Die Abfragezeichenfolge wird segmentiert, analysiert und im Index gesucht.

Sowohl die Analyse des Dokuments als auch die der Abfragezeichenfolge lassen sich wie folgt unterteilen:

- Basisunterstützung, die nicht auf einem Wörterverzeichnis basiert. Hierzu gehören Leerraumsegmentierung und N-Gram-Segmentierung.
- Linguistische Unterstützung, die auf einem Wörterverzeichnis basiert. Hierzu gehören die Wort- und Satzsegementierung und die Reduktion auf die Grundform.

  Zur Verarbeitung auf linguistischer Basis gehört die lexikalische Analyse, ein Prozess, bei dem alternative Darstellungen des Eingabetexts erstellt und den im Eingabetext erkannten Token alle verfügbaren Verzeichnisdaten zugeordnet werden. Durch die Verwendung der erweiterten Sprachenverarbeitung wurde die Suchqualität sehr verbessert.

  Zugehörige Konzepte

  „Spracherkennung"
  Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

  „Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung" auf Seite 68
  Für Dokumente, die in Sprachen abgefasst sind, die nicht von den Erkennung der Sprache und der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Leerraumsegmentierung und N-Gram-Segmentierung zur Verfügung.

Spracherkennung

Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

Die Unternehmenssuche erkennt die folgenden Sprachen automatisch:

```
Arabisch                                 Französisch Koreanisch
Chinesisch (traditionell und vereinfacht) Deutsch     Polnisch
Tschechisch                              Griechisch  Portugiesisch
Dänisch                                  Hebräisch   Russisch
Niederländisch                           Ungarisch   Spanisch
Englisch                                 Italienisch Schwedisch
Finnisch                                 Japanisch   Türkisch
```

Die linguistischen Prozesse in der Unternehmenssuche ermitteln die Sprache eines Quellendokuments während der Indexierung, nicht während der Abfrageverarbeitung.

In der Unternehmenssuche können Sie die automatische Spracherkennung für ein Dokument angeben, Sie können aber auch eine Sprache auswählen, die verwendet werden soll.

Wenn Sie die automatische Spracherkennung ausgewählt haben, der Parser die Sprache des Dokuments jedoch nicht ermitteln kann, verwendet der Parser die Sprache, die Sie beim Erstellen des Crawlers in der Administrationskonsole für die Unternehmenssuche angeben.

Wenn Sie keine automatische Spracherkennung auswählen, wird immer die von Ihnen angegebene Sprache verwendet. Der Standardwert ist Englisch.

Dokumente, für die es keine sprachspezifischen Wörterverzeichnisse gibt, werden unter Verwendung einer sprachunabhängigen Basistechnologie verarbeitet, wie z. B. Leerraumsegmentierung und N-Gram-Segmentierung.

Die Spracherkennung der Unternehmenssuche ist für einsprachige Dokumente bestens geeignet. Bei mehrsprachigen Dokumenten wird versucht zu ermitteln, welche Sprache im Dokument am häufigsten verwendet wird. Die Analyseergebnisse sind jedoch nicht immer zufriedenstellend.

Anhand der Sprache eines Dokuments können Sie Ihre Suchergebnisse so beschränken, dass nur Dokumente angezeigt werden, die in einer bestimmten Sprache abgefasst sind. Wenn Sie z. B. nach Dokumenten suchen, die über Jacques Chirac geschrieben wurden, können Sie angeben, dass nur auf Französisch verfasste Dokumente in die Suchergebnisse aufgenommen werden sollen.

Zugehörige Konzepte

„Textanalyse innerhalb der Unternehmenssuche" auf Seite 67
Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

„Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung"
Für Dokumente, die in Sprachen abgefasst sind, die nicht von den Erkennung der Sprache und der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Leerraumsegmentierung und N-Gram-Segmentierung zur Verfügung.

Linguistische Unterstützung der nicht wörterverzeichnisbasierten Segmentierung

Für Dokumente, die in Sprachen abgefasst sind, die nicht von den Erkennung der Sprache und der lexikalischen Analysetechnologie unterstützt werden, stellt die Unternehmenssuche eine Basisunterstützung in der Form von Unicode-basierter Leerraumsegmentierung und N-Gram-Segmentierung zur Verfügung.

Unicode-basierte Leerraumsegmentierung
Dieses Verarbeitungsverfahren auf linguistischer Basis verwendet den Leerraum (oder die Leerzeichen) zwischen Wörtern als Wortbegrenzer.

N-Gram-Segmentierung
Dieses Verarbeitungsverfahren auf linguistischer Basis behandelt überlap-

pende Sequenzen von *n* Zeichen als ein Wort. Dieses einfache Segmentierungsverfahren ist für viele Abruftasks ausreichend.

Diese Verfahren sind unabhängig von Wörterverzeichnissen in bestimmten Sprachen. Sie enthalten keine hoch entwickelte Technologie für die Verarbeitung auf linguistischer Basis, wie die Reduktion auf die Grundform.

Die N-Gram-Segmentierung wird für Sprachen wie Thailändisch verwendet, in denen es keine Leerstellen gibt, die als Begrenzer verwendet werden können. Dasselbe Verfahren wird für Hebräisch und Arabisch angewendet. Obwohl es in diesen zwei Sprachen Leerzeichen als Begrenzer vorhanden sind, gibt die N-Gram-Segmentierung bessere Ergebnisse zurück als die Basisform der Unicode-basierten Leerraumsegmentierung.

Zugehörige Konzepte

„Textanalyse innerhalb der Unternehmenssuche" auf Seite 67
Zu der in der Unternehmenssuche enthaltenen Textanalyse gehört die Erkennung der Sprache und die Segmentierung des Dokuments.

„Spracherkennung" auf Seite 67
Bevor eine Wort- und Satzsegmentierung, eine Zeichennormalisierung oder eine Reduktion auf die Grundform ausgeführt werden kann, muss die Unternehmenssuche die Sprache des Quellendokuments ermitteln.

Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen

Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

Die Segmentierung ist der Prozess, bei dem der Eingabetext in eindeutige lexikalische Texteinheiten unterteilt wird. Dazu gehören einige der folgenden Verarbeitungsmethoden auf linguistischer Basis:

Wortsegmentierung

Die Wortsegmentierung wird für Sprachen verwendet, die zwischen Wörtern keinen Leerraum (oder keine Begrenzer) verwenden, wie z. B. Japanisch und Chinesisch.

Reduktion auf die Grundform

Die Reduktion auf die Grundform ist eine Verarbeitungsform auf linguistischer Basis, bei der für jedes Wort, das im Text vorkommt, das Lemma ermittelt wird. Das *Lemma* eines Worts umfasst dessen Grundform sowie flektierte Formen derselben Wortart. So umfasst z. B. das Lemma von *gehen* die Formen *gehen*, *geht*, *ging*, *gegangen* und *gehend*. Lemmata von Substantiven gruppieren die Formen von Singular und Plural (wie *Kalb* und *Kälber*). Lemmata von Adjektiven gruppieren die Formen von Komparativ und Superlativ (wie *gut*, *besser* und *am besten*). Lemmata von Pronomen gruppieren den Kasus eines Pronomens (wie *ich*, *mich*, *mein* und *mir*).

Für die Reduktion auf die Grundform ist sowohl zum Indexieren als auch zum Suchen ein Wörterverzeichnis erforderlich.

Die Unternehmenssuche indexiert die Lemmata und flektierten Wörter und reduziert alle flektierten Wörter in einer Abfrage auf ihre Grundform. Durch die Reduktion auf die Grundform wird die Suchqualität verbessert, indem nach Dokumenten gesucht wird, die Varianten eines in der Abfrage

enthaltenen flektierten Worts aufweisen. So wird z. B. auch nach Dokumenten gesucht, in denen das Wort *Mäuse* vorkommt, wenn die Abfrage das Wort *Maus* enthält.

Kontraktionssplitting
Die Suchqualität wird verbessert, indem Kontraktionen erkannt und in ihre einzelnen Komponenten gesplittet werden. Beispiel:

wouldn't wird gesplittet in
would + not
Horse's wird gesplittet in *Horse + is* oder *'s*
(um der Mehrdeutigkeit einer Abfrage Rechnung zu tragen)

Klitikerkennung
Klitika sind eine besondere Form der Kontraktion, und die Suchqualität wird verbessert, wenn ihre einzelnen Komponenten ermittelt werden. Ein *Klitik* ist ein Element, das sich wie ein Affix und ein Wort verhält. Klitika sind jedoch schwer zu ermitteln, da sie auch Teil der Wortbildung sind. Anders als andere morphologische Phänomene (Wortstrukturen) treten Klitika in einer syntaktischen Struktur auf, und ihr Anschluss an das Wort ist nicht Teil der Wortbildungsregeln. Beispiel:

reparti-lo-emos hat die Komponenten *repartir + lo + emos*
l'avenue hat die Komponenten *le + avenue*
dell'arte hat die Komponenten *dello + arte*.

Erkennung nicht alphabetischer Zeichen
Die Verarbeitungsprozesse auf linguistischer Basis erkennen nicht alphabetische Zeichen. Abhängig von der internen sprachabhängigen Logik, werden einige nicht alphabetische Zeichen als eigene lexikalische Einheiten unterschiedlichen Typs zurückgegeben, andere werden in Gruppen zusammengefasst.

So werden bei Klitika z. B. Hochkommata oder Silbentrennungsstriche als Teil des Worts betrachtet, und bei unbekannten Abkürzungen werden Sie wie ein Punkt behandelt. Die Verarbeitung auf linguistischer Basis kann auch einige bestimmte Zeichenfolgen als Token erkennen, z. B. URLs, E-Mail-Adressen und Datumsangaben.

Erkennung von Abkürzungen
Die Verarbeitung auf linguistischer Basis erkennt Abkürzungen, die im Wörterverzeichnis als eine lexikalische Einheit angegeben sind. Ist eine Abkürzung nicht im Wörterverzeichnis vorhanden, wird sie zwar als lexikalisches Element erkannt, es werden ihr jedoch keine Informationen aus dem Wörterverzeichnis zugeordnet.

Das korrekte Erkennen von Abkürzungen ist wichtig für die Satzerkennung. So ist z. B. der Punkt hinter einer Abkürzung nicht unbedingt das Ende eines Satzes.

Erkennung der Markierung des Satzendes
Für die Satzsegmentierung können die Verarbeitungsprozesse auf linguistischer Basis die Markierungen des Satzendes richtig erkennen.

Linguistische Unterstützung auf Basis von Wörterverzeichnissen ist für die folgenden Sprachen verfügbar:

Arabisch	Italienisch
Chinesisch (traditionell und vereinfacht)	Japanisch
Tschechisch	Koreanisch
Dänisch	Norwegisch (Bokmal und Nynorsk)

```
Niederländisch                          Polnisch
Englisch                                Portugiesisch (Portugal und Brasilien)
Finnisch                                Russisch
Französisch (Frankreich und Kanada)     Spanisch
Deutsch (Deutschland und Schweiz)       Schwedisch
Griechisch
```

Zugehörige Konzepte

„Wortsegmentierung im Japanischen"
Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

„Orthografische Varianten im Japanischen"
Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Wortsegmentierung im Japanischen

Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

Ein Beispiel für diese Optimierung ist die Zerlegung von Wörtern. Im Japanischen werden viele zusammengesetzte Begriffe verwendet. Diese Begriffe werden in Token von einer optimalen Größe zerlegt, so dass bessere Suchergebnisse erzielt werden. Flektierte Wörter und Präpositionen werden ebenfalls zerlegt, um die Suchleistung zu verbessern.

Zugehörige Konzepte

„Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen" auf Seite 69
Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

„Orthografische Varianten im Japanischen"
Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Orthografische Varianten im Japanischen

Im Japanischen werden viele orthografischen Varianten verwendet. Die Katakana-Varianten sind am wichtigsten, da Katakana häufig verwendet wird, um fremdsprachige Wörter zu buchstabieren und auszusprechen. Viele der Katakana-Varianten werden häufig im Japanischen verwendet.

Die Unternehmenssuche verwendet ein Wörterverzeichnis von Varianten, um die typischen Katakana-Varianten ihren Basisformen (ähnlich wie ein Lemma) zuzuordnen, so dass alle Dokumente, einschließlich derer mit orthografischen Varianten des Katakana-Worts in der Abfragezeichenfolge, gefunden werden.

Die Unternehmenssuche unterstützt außerdem auch die typischen Okurigana-Varianten, bei denen es sich um Kanji-Wortendungen handelt, die in Hiragana geschrieben werden.

Zugehörige Konzepte

„Linguistische Unterstützung der Segmentierung auf der Basis von Wörterverzeichnissen" auf Seite 69
Wenn die Sprache eines Dokuments richtig erkannt wurde und sprachspezifische Wörterverzeichnisse verfügbar sind, wird die entsprechende Verarbeitung auf linguistischer Basis angewendet.

„Wortsegmentierung im Japanischen" auf Seite 71
Wird ein Textdokument oder eine Abfragezeichenfolge als Japanisch erkannt, führt die Unternehmenssuche die relevante Wortsegmentierung aus, indem sie die für die japanische Sprache optimierte morphologische Analysetechnologie verwendet.

Stoppwortentfernung

In der Unternehmenssuche werden alle Stoppwörter, z. B. häufig verwendete Wörter wie *ein* und *der*, aus Mehrwortabfragen entfernt, um die Suchleistung zu optimieren.

Die Stoppworterkennung im Japanischen basiert auf grammatikalischen Informationen. So erkennt die Unternehmenssuche z. B., ob es sich bei einem Wort um ein Substantiv oder ein Verb handelt, wo für andere Sprachen Speziallisten verwendet werden.

Zugehörige Konzepte

„Zeichennormalisierung"
Die Zeichennormalisierung ist ein Prozess, der das Abrufen verbessern kann. Abrufverbesserung durch Zeichennormalisierung bedeutet, dass mehr Dokument abgerufen werden, selbst wenn die Dokument nicht genau mit der Abfrage übereinstimmen.

Zeichennormalisierung

Die Zeichennormalisierung ist ein Prozess, der das Abrufen verbessern kann. Abrufverbesserung durch Zeichennormalisierung bedeutet, dass mehr Dokument abgerufen werden, selbst wenn die Dokument nicht genau mit der Abfrage übereinstimmen.

Die Unternehmenssuche verwendet eine Unicode-kompatible Normalisierung, zu der auch die Normalisierung asiatischer Zeichen mit voller Breite und halber Breite gehört.

So wird z. B. im Japanischen ein alphanumerisches Zeichen mit voller Breite zum Zeichen mit halber Breite normalisiert, ein Katakana-Zeichen mit halber Breite zum Zeichen mit voller Breite und so weiter. Die Unternehmenssuche entfernt auch die mittleren Punkte aus Katakana-Schriftzeichen, die im Japanischen als Begrenzer von zusammengesetzten Begriffen verwendet werden.

Die anderen Formen der Zeichennormalisierung umfassen Folgendes:

Normalisierung von Groß-/Kleinschreibung
Damit z. B. Dokumente mit *USA* angezeigt werden, wenn Sie nach *usa* suchen.

Umlautauflösung
Damit z. B. Dokumente mit *schoen* angezeigt werden, wenn Sie nach *schön* suchen.

Entfernen von Akzenten
Damit z. B. Dokumente mit *é* angezeigt werden, wenn Sie nach *e* suchen.

Entfernen anderer diakritischer Zeichen
Damit z. B. Dokumente mit *ç* angezeigt werden, wenn Sie nach *c* suchen.

Ligaturauflösung
Damit z. B. Dokumente mit *Æ* angezeigt werden, wenn Sie nach *ae* suchen.

Alle Normalisierungen funktionieren in beide Richtungen. Sie finden auch Dokumente mit *usa*, wenn Sie nach *USA* suchen, und Dokumente, die Wörter mit *e* enthalten, wenn Sie nach *é* suchen und so weiter. Sie können die Normalisierungen auch kombinieren. Sie finden z. B. auch Dokumente, die *météo* enthalten, wenn Sie nach *METEO* suchen.

Die Normalisierungen basieren auf den Unicode-Zeichenmerkmalen und sind nicht sprachenabhängig. So unterstützt die Unternehmenssuche z. B. das Entfernen diakritischer Zeichen aus dem Hebräischen und die Ligaturauflösung im Arabischen.

Zugehörige Konzepte

„Stoppwortentfernung" auf Seite 72
In der Unternehmenssuche werden alle Stoppwörter, z. B. häufig verwendete Wörter wie *ein* und *der*, aus Mehrwortabfragen entfernt, um die Suchleistung zu optimieren.

Dokumentation zur Unternehmenssuche

Die Dokumentation zu WebSphere Information Integrator OmniFind Edition steht im PDF- oder HTML-Format zur Verfügung.

Das WebSphere Information Integrator OmniFind Edition-Installationsprogramm kann die Installation der Informationszentrale automatisch ausführen. Das Installationsprogramm installiert die Informationszentrale auf dem Suchserver. Bei einer Installation auf mehreren Servern wird die Informationszentrale auf beiden Suchservern installiert. Wenn Sie die Informationszentrale nicht installieren, wird beim Anklicken von **Hilfe** die Informationszentrale auf einer IBM Website geöffnet. Starten Sie die Informationszentrale, um HTML-Themen für die Unternehmenssuche anzuzeigen.

Wechseln Sie in das Verzeichnis `docs/`*`ländereinstellung`*`/pdf`, um die PDF-Dokumente anzuzeigen. Wenn Sie beispielsweise Dokumente in englischer Sprache suchen, wechseln Sie in das Verzeichnis `docs/en_US/pdf`. Sie haben auch die Möglichkeit, die PDF-Dokumentation sowie Downloads, Korrekturen, technische Hinweise und die Informationszentrale von der WebSphere Information Integrator OmniFind Edition-Unterstützungssite aus anzuzeigen.

In der folgenden Tabelle ist die verfügbare Dokumentation mit Dateinamen und Speicherposition aufgeführt.

Tabelle 9. PDF-Dokumentation zur Unternehmenssuche

Dokument	Dateiname	Speicherposition
Installationshandbuch für die Unternehmenssuche (Themen zu diesem Dokument stehen auch in der Informationszentrale zur Verfügung)	`iiysi.pdf`	`docs/`*`ländereinstellung`*`/pdf/`
Verwaltung der Unternehmenssuche (Themen zu diesem Dokument stehen auch in der Informationszentrale zur Verfügung.)	`iiysa.pdf`	`docs/`*`ländereinstellung`*`/pdf/`
Programming Guide and API Reference for Enterprise Search (Themen zu diesem Dokument stehen auch in der Informationszentrale zur Verfügung.)	`iiysp.pdf`	`docs/`*`ländereinstellung`*`/pdf/`
Fehlernachrichten (Themen zu diesem Dokument stehen auch in der Informationszentrale zur Verfügung.)	`iiysm.pdf`	`docs/`*`ländereinstellung`*`/pdf/`
Installation Requirements for Enterprise Search (Themen zu diesem Dokument stehen auch in der Informationszentrale zur Verfügung.)	`iiysr.txt` oder `iiysr.htm`	`docs/`*`ländereinstellung`*`/` (Diese Datei kann auch über das Programm **First Steps** aufgerufen werden.)

Tabelle 9. PDF-Dokumentation zur Unternehmenssuche (Forts.)

Dokument	Dateiname	Speicherposition
Release-Informationen	`iiysn.pdf`	Ist nur auf der IBM WebSphere Information Integrator OmniFind Edition Documentation-Website verfügbar.
Integration der Textanalyse	`iiyst.pdf`	`docs/`*`ländereinstellung`*`/pdf/`

WebSphere II OmniFind Edition - Behindertengerechte Bedienung

Die IBM WebSphere Information Integrator OmniFind Edition-Benutzerschnittstellen und die zugehörige Dokumentation sind für behindertengerechte Bedienung geeignet.

Installationsprogramm

Sie können mit Hilfe von Direktaufrufen über die Tastatur im WebSphere II OmniFind Edition-Installationsprogramm navigieren. Die folgende Tabelle enthält eine Beschreibung einiger Direktaufrufe über die Tastatur.

Tabelle 10. Direktaufrufe über die Tastatur für das Installationsprogramm

Aktion	**Direktaufruf**
Hervorheben eines Radioknopfs	Pfeiltaste
Auswählen eines Radioknopfs	Tabulatortaste
Hervorheben eines Druckknopfs	Tabulatortaste
Auswählen eines Druckknopfs	Eingabetaste
Wechseln zum nächsten oder vorhergehenden Fenster oder Ausführen eines Abbruchs	Heben Sie einen Druckknopf durch Drücken der Tabulatortaste hervor, und drücken Sie die Eingabetaste.
Inaktivieren des aktiven Fensters	Strg + Alt + Esc

Administrationskonsole für die Unternehmenssuche und Informationszentrale

Die Administrationskonsole und die Informationszentrale sind browserbasierte Schnittstellen, die sich über Microsoft Internet Explorer oder Mozilla FireFox anzeigen lassen. Eine Liste der Direktaufrufe über die Tastatur und andere Funktionen zur behindertengerechten Bedienung von Internet Explorer bzw. FireFox finden Sie in der Onlinehilfe des jeweiligen Browsers.

PDF-Dokumentation

Die gesamte Dokumentation zur Unternehmenssuche liegt im PDF-Format vor. Sie können über Adobe Acrobat Version 6.0 auf die PDF-Dokumente zugreifen. Die PDF-Dokumente sind strukturiert und sollten von den meisten Sprachausgabeprogramme gelesen werden können.

Glossar der Begriffe für die Unternehmenssuche

Dieses Glossar enthält Begriffe, die in den Schnittstellen für die Unternehmenssuche und in der zugehörigen Dokumentation verwendet werden.

Abfrage in natürlicher Sprache (Natural Language Query)
Eine Art der Suche, bei der geschriebene Ausdrücke (z. B. ″Wer leitet die Finanzabteilung?″) und nicht eine einfache Gruppe von Schlüsselwörtern analysiert werden.

Abschließendes Zeichen (Trailing Character)
Ein Zeichen an der letzten Position in einem Wort.

Administrator für die Unternehmenssuche (Enterprise Search Administrator)
Eine Verwaltungsrolle, mit der ein Benutzer das gesamte System für die Unternehmenssuche verwalten kann.

Allgemeine Analysestruktur (Common Analysis Structure)
Eine Struktur, in der ein Dokument gespeichert wird, das von einer Textanalysesteuerkomponente analysiert wird. Die Informationen werden in einer allgemeinen Analysestruktur in Form von Annotationen und anderen Merkmalstrukturen gespeichert.

Allgemeine Übertragungsschicht (CCL - Common Communication Layer)
Die Kommunikationsinfrastruktur, die die verschiedenen Komponenten (Controller, Parser, Crawler, Indexierungskomponente) von WebSphere Information Integrator OmniFind Edition miteinander verbindet.

Analyseergebnisse (Analysis Results)
Die von Annotatoren erstellten Informationen. Analyseergebnisse, die den von Ihnen gesuchten Informationen entsprechen, werden in eine Datenstruktur geschrieben, die allgemeine Analysestruktur genannt wird.

Annotation
Informationen zu einer Textpassage. Eine Annotation (ergänzender Kommentar) könnte beispielsweise darauf hinweisen, dass eine kurze Textpassage für einen Unternehmensnamen steht. In UIMA ist eine Annotation eine besondere Merkmalstruktur.

Annotator
Eine Softwarekomponente, die bestimmte linguistische Analysetasks ausführt und Annotationen (ergänzende Kommentare) erstellt und erfasst. Der Annotator (Kommentatorfunktion) ist die Analyselogikkomponente einer Analysesteuerkomponente.

Archiv der Verarbeitungssteuerkomponente (Processing Engine Archive)
Eine komprimierte .pear-Archivdatei, die eine UIMA-Analysesteuerkomponente sowie sämtliche Ressourcen enthält, die zu ihrer Nutzung für eine benutzerdefinierte Analyse in der Unternehmenssuche erforderlich sind.

Aufspürfunktion (Discoverer)
Eine Crawlerfunktion, die feststellt, welche Datenquellen dem Crawler zum Abrufen von Informationen zur Verfügung stehen.

Aus Warteschlange entfernen (Dequeue)
Einträge aus einer Warteschlange entfernen.

Ausgangs-URL (Seed URL)
Der Ausgangspunkt einer Crawlersuche.

Bediener (Operator)
Ein Benutzer der Unternehmenssuche, der über die Berechtigung zum Beobachten, Starten und Stoppen von Prozessen auf Objektgruppenebene verfügt.

Begriffsextraktion (Concept Extraction)
Eine Suchfunktion, die signifikante Vokabularelemente (z. B. Personen, Orte oder Produkte) in Textdokumenten identifiziert und eine Liste dieser Elemente erstellt. Siehe auch Themenextraktion.

Benutzeragent (User Agent)
Eine Anwendung, die das Internet durchsucht und Informationen zu sich selbst auf den besuchten Sites hinterlässt. In der Unternehmenssuche ist der Web-Crawler ein Benutzeragent.

Berechtigungsnachweis (Credential)
Während der Authentifizierung zugewiesene detaillierte Informationen, die den Benutzer, gegebenenfalls vorhandene Gruppenzuordnungen und andere sicherheitsrelevante Identitätsattribute beschreiben. Mit Berechtigungsnachweisen lassen sich eine Vielzahl von Services ausführen, wie z. B. Berechtigung, Prüfung und Delegierung.

Bereich (Place)
Ein virtueller Ort, der im Portal angezeigt wird und in dem sich Einzelpersonen sowie Gruppen zur Zusammenarbeit online treffen. In einem Portal verfügt jeder Benutzer über einen eigenen Bereich für seine persönliche Aufgaben und darüber hinaus haben Einzelpersonen und Gruppen Zugang zu einer Reihe gemeinsam genutzter Bereiche, die allgemein zugängliche oder eingeschränkte Bereiche sein können. Siehe auch Lotus QuickPlace-Bereich.

Bereich (Scope)
Eine Gruppe zusammengehöriger URIs, mit denen der Bereich einer Suchanforderung definiert wird.

Bibliothek (Library)
Ein Systemobjekt, das anderen Objekten als Verzeichnis dient. Siehe auch Domino Document Manager-Bibliothek.

Boostklasse (Boost Class)
Eine Angabe, mit der der relative Rang eines Dokuments in den Suchergebnissen beeinflusst werden kann.

Boostwort (Boost Word)
Ein Wort, mit dem der relevante Rang eines Dokuments in den Suchergebnissen beeinflusst werden kann. Bei der Abfrageverarbeitung erhält ein Dokument, das ein Boostwort enthält, möglicherweise einen höheren oder niedrigeren Rang, je nachdem welche Bewertung für das Wort vordefiniert wurde.

Crawler
Ein Softwareprogramm, das Dokumente aus Datenquellen abruft und Informationen zusammenstellt, mit denen Suchindizes erstellt werden können.

Crawlerbereich (Crawl Space)
Eine bestimmten Mustern (wie Datenbanknamen, Dateisystempfaden,

Domänennamen, IP-Adressen und URLs) entsprechende Gruppe von Quellen, die ein Crawler liest, um Elemente zum Indexieren abzurufen.

Datenquelle (Data Source)
Jedes Datenrepository, aus dem Dokumente abgerufen werden können, z. B. das Internet, relationale und nicht relationale Datenbanken sowie Content-Management-Systeme.

Datenquellentyp (Data Source Type)
Eine Zusammenfassung von Datenquellen nach dem Protokoll, mit dem auf die Daten zugegriffen wird.

Definierter Name (Distinguished Name)
Der Name, der einen Eintrag in einem Verzeichnis eindeutig identifiziert. Ein definierter Name besteht aus durch Kommata getrennten Attribut:Wert-Paaren. Außerdem eine Gruppe von Name/Wert-Paaren (z. B. CN=Name der Person und C=Land oder Region), die eine Entität in einem digitalen Zertifikat eindeutig identifizieren.

Dokumentobjektmodell (Document Object Model)
Ein System, bei dem ein gegliedertes Dokument (z. B. eine XML-Datei) in Form einer Baumstruktur mit Objekten angezeigt wird, auf die über das Programm zugegriffen werden kann und die aktualisierbar sind.

Domino Document Manager-Aktenschrankdatei (Domino Document Manager Cabinet)
Eine Domino Document Manager-Datenbank, die zum Organisieren von Dokumenten verwendet wird. Aktenschrankdateien enthalten Domino-Datenbanken.

Domino Document Manager-Bibliothek (Domino Document Manager Library)
Eine Domino Document Manager-Datenbank, die den Einstiegspunkt in Domino Document Manager bildet.

Domino Internet Inter-ORB Protocol (DIIOP)
Eine Server-Task, die auf dem Server ausgeführt wird und mit dem Domino Object Request Broker zusammenarbeitet, um eine Kommunikation zwischen Java-Applets zu ermöglichen, die mit Notes-Java-Klassen und mit dem Domino-Server erstellt werden. Browserbenutzer und Domino-Server führen die Kommunikation und den Austausch von Objektdaten über DIIOP aus.

Dynamische Rangfolge (Dynamic Ranking)
Ein Rangfolgetyp, bei dem die Begriffe in der Abfrage in Hinblick auf die durchsuchten Dokumente analysiert werden, um die Rangfolge der Ergebnisse zu ermitteln. Siehe auch textbasierte Bewertung. Vergleiche statische Rangfolge.

Dynamische Zusammenfassung (Dynamic Summarization)
Eine Art der Zusammenfassung, bei der die Suchbegriffe hervorgehoben werden und die Suchergebnisse Ausdrücke enthalten, die die Konzepte des gesuchten Dokuments am besten darstellen. Vergleiche statische Zusammenfassung.

Escapezeichen (Escape Character)
Ein Zeichen, das eine spezielle Bedeutung für mindestens ein nachfolgendes Zeichen unterdrückt oder auswählt.

Externe Datenquellen (External Data Source)
Eine Datenquelle für die Föderation, die nicht von WebSphere Information Integrator OmniFind Edition durchsucht, syntaktisch analysiert oder inde-

xiert wird. Das Durchsuchen von externen Datenquellen wird an die Anwendungsprogrammierschnittstelle für die Abfrage dieser Datenquellen delegiert.

Feld (Field)
Der kleinste identifizierbare Teil eines Datensatzes.

Feldspezifische Suche (Fielded Search)
Eine auf ein bestimmtes Feld beschränkte Abfrage.

Ferner Föderator (Remote Federator)
Ein Serverföderator, der eine Föderation für eine Gruppe durchsuchbarer Objekte ausführt.

Freiformatsuche (Free Text Search)
Eine Suche, in der der Suchbegriff als unformatierter Text dargestellt wird.

Föderation (Federation)
Der Prozess des Kombinierens von Benennungssystemen, wodurch es dem zusammengefassten System ermöglicht wird, zusammengesetzte Namen zu verarbeiten, die alle Benennungssysteme umfassen.

Föderierte Suche (Federated Search)
Eine Suchfunktionalität, die das Durchsuchen mehrerer Suchservices ermöglicht und eine konsolidierte Liste mit Suchergebnissen zurückgibt.

Hybridsuche (Hybrid Search)
Eine Kombination aus Boolescher Suche und Freiformatsuche.

Identitätsmanagement (Identity Management)
Eine Funktionalität zur Verschlüsselung von Benutzerberechtigungen in einem sicheren Speicher.

In Warteschlange stellen (Enqueue)
Einträge in eine Warteschlange einfügen.

Index Siehe Volltextindex.

Indexaktualisierung (Index Refresh)
Das Hinzufügen neuer Informationen zu einem vorhandenen Index in einem System für die Unternehmenssuche. Vergleiche Indexreorganisation.

Indexierungswarteschlange (Index Queue)
Eine Liste von zu verarbeitenden Anforderungen zur Indexreorganisation oder zur Indexaktualisierung.

Indexreorganisation (Index Reorganization)
Das Aufbauen des Index in einem System für die Unternehmenssuche. Vergleiche Indexaktualisierung.

Informationsextraktion (Information Extraction)
Eine Art der Begriffsextraktion, bei der signifikante Vokabularelemente (z. B. Namen, Begriffe und Ausdrücke) in Textdokumenten automatisch erkannt werden.

IP-Adresse (IP Address)
Eine eindeutige 32-Bit-Adresse, die einen Host im Netz identifiziert.

Java Database Connectivity (JDBC)
Ein Industriestandard für datenbankunabhängige Konnektivität zwischen der Java-Plattform und einer großen Reihe von Datenbanken. Die JDBC-Schnittstelle bietet eine API auf Aufrufebene für SQL-Datenbankzugriffe.

Java Virtual Machine (JVM)
Softwareimplementierung eines Prozessors, die kompilierten Java-Code (Applets und Anwendungen) ausführt.

JavaScript
Eine Web-Scripting-Sprache, die in Browsern und auf Web-Servern verwendet wird.

JavaServer Pages (JSP)
Eine Servertechnologie zur Scripterstellung, die es ermöglicht, Java-Code dynamisch in Webseiten (HTML-Dateien) einzubetten und diese auszuführen, wenn die Seite bereitgestellt wird, um einem Client dynamischen Inhalt zurückzugeben.

Katakana
Ein Zeichensatz aus Symbolen, die in einem der beiden gebräuchlichen phonetischen Alphabete der japanischen Sprache verwendet werden. Dieser dient in erster Linie zum phonetischen Schreiben von Fremdwörtern.

Kategorie (Category)
Eine Gruppe von Dokumenten mit ähnlichen Merkmalen.

Kategoriebaum (Category Tree)
Eine Kategoriehierarchie, die an der Administrationskonsole für die Unternehmenssuche angezeigt wird.

Klitik (Clitic)
Ein Wort, das syntaktisch eigenständig ist, phonetisch aber mit einem anderen Wort zusammenhängt. Ein Klitik kann mit dem Wort, an das es angelehnt ist, zusammengeschrieben oder davon getrennt geschrieben werden. Typische Beispiele für in der englischen Sprache vorkommende Klitika sind der hintere Teil einer Zusammenfügung (*would**n't*** oder *you**'re***).

Komponentenpfad (Feature Path)
Ein Pfad, über den auf den Wert eines Merkmals in einer UIMA-Merkmalstruktur zugegriffen wird.

Lemma
Die kanonische Form eines Worts. Lemmas spielen vor allem in stark flektierten Sprachen wie dem Tschechischen eine wichtige Rolle.

Lexikalische Affinität (Lexical Affinity)
Die Beziehung von Suchbegriffen, die eine ähnliche Bedeutung im Dokument haben. Mit der lexikalischen Affinität wird die Relevanz eines Ergebnisses berechnet.

Ligatur (Ligature)
Mindestens zwei Zeichen, die so miteinander verbunden werden, dass sie ein einzelnes Zeichen bilden (z. B. das Verbinden von f und i zur Ligatur fi).

Lightweight Directory Access Protocol (LDAP)
Ein offenes Protokoll, das mit Hilfe von TCP/IP Zugriff auf Verzeichnisse ermöglicht, die ein X.500-Modell unterstützen, und das nicht die Ressourcenanforderungen des komplexeren X.500 Directory Access Protocol aufweist.

Linguistische Suche (Linguistic Search)
Eine Art der Suche, bei der ein Dokument mit auf ihre Grundformen reduzierten Begriffen durchsucht, abgerufen und indexiert (Beispiel: *Mäuse* wird als *Maus* indexiert) oder mit ihrer Grundform erweitert wird (wie bei zusammengesetzten Wörtern).

Linkanalyse (Link Analysis)
Ein Verfahren, das auf der Analyse von Hyperlinks zwischen Dokumenten basiert und mit dem festgestellt wird, welche Seiten in der Objektgruppe für Benutzer von Bedeutung sind.

Lokaler Föderator (Local Federator)
Ein Clientföderator, der eine Gruppe durchsuchbarer Objekte föderiert.

Lotus QuickPlace-Bereich (Lotus QuickPlace Place)
Ein von Lotus QuickPlace bereitgestellter Arbeitsbereich im Web, der geographisch weit verteilten Teilnehmern die Möglichkeit bietet, zusammen an Projekten zu arbeiten und online in einem strukturierten und sicheren Arbeitsbereich miteinander zu kommunizieren.

Lotus QuickPlace-Raum (Lotus QuickPlace Room)
Ein partitionierter Bereich in einem Lotus QuickPlace-Bereich, der ausschließlich berechtigten Mitgliedern vorbehalten ist, die eine gemeinsame Aufgabe und die Notwendigkeit zur Zusammenarbeit verbindet.

Merkmalstruktur (Feature Structure)
Die zu Grunde liegende Datenstruktur, die dem Ergebnis der Textanalyse entspricht. Die Merkmalstruktur hat eine Attribut-Wert-Struktur. Jede Merkmalstruktur ist von einem bestimmten Typ, wobei jeder Typ, ähnlich wie eine Java-Klasse, über eine angegebene Gruppe gültiger Merkmale oder Attribute verfügt.

MIME-Typ (MIME Type)
Ein Internetstandard zur Angabe des Typs eines Objekts, das über das Internet übertragen wird.

Modellbasierte Kategorie (Model-Based Category)
Eine Taxonomie mit vordefinierten Begriffen, mit der das Thema eines Dokuments bestimmt wird, damit das Dokument zusammen mit Dokumenten ähnlichen Inhalts indexiert und durchsucht werden kann.

N-Gram-Segmentierung (N-Gram Segmentation)
Eine Analysemethode, bei der nicht wie bei der Unicode-basierten Leerzeichensegmentierung Wörter durch eine Leerstelle begrenzt sind, sondern sich überlappende Folgen einer bestimmten Anzahl Zeichen als ein Wort betrachtet werden.

No-Follow-Anweisung (No-Follow Directive)
Eine Anweisung in einer Webseite, die Roboter (z. B. den Web-Crawler) anweisen, den Links auf diesen Seiten nicht zu folgen.

No-Index-Anweisung (No-Index Directive)
Eine Anweisung in einer Webseite, die Roboter (z. B. den Web-Crawler) anweisen, den Inhalten auf diesen Seiten nicht in den Index einzuschließen.

Notes Remote Procedure Call (NRPC)
Die für die gesamte Notes-zu-Notes-Kommunikation verwendete architekturelle Schicht von Lotus Notes.

Objektgruppe (Collection)
Eine Gruppe von Datenquellen und Optionen für die Crawlersuche, die Syntaxanalyse, das Indexieren und das Durchsuchen dieser Datenquellen.

Parametrische Suche (Parametric Search)
Eine Art der Suche, bei der Objekte gesucht werden, die einen numeri-

schen Wert oder ein numerisches Attribut (wie z. B. Datumsangaben, ganze Zahlen oder andere numerische Datentypen in einem angegebenen Bereich) enthalten.

Parser Ein Programm, das Dokumente interpretiert, die dem Datenspeicher für die Unternehmenssuche hinzugefügt werden. Der Parser extrahiert Informationen aus den Dokumenten und bereitet sie für Indexierungs-, Such- und Abrufvorgänge vor.

Platzhalterzeichen (Masking Character)
Ein Zeichen, das optionale Zeichen am Anfang, in der Mitte und am Ende eines Suchbegriffs darstellt. Mit Platzhalterzeichen werden normalerweise Varianten eines Begriffs in einem Index gesucht.

Platzhalterzeichen (Wildcard Character)
Ein Zeichen, das optionale Zeichen am Anfang, in der Mitte oder am Ende eines Suchbegriffs darstellt.

Popularitätsrangfolge (Popular Ranking)
Ein Rangfolgetyp, der die vorhandene Rangfolge eines Dokuments gemäß der Popularität des Dokuments ergänzt.

Protokoll zum Sperren von Websitebereichen für Robots (Robots Exclusion Protocol) Ein Protokoll mit dem Website-Administratoren durchsuchende Robots anweisen können, welche Bereiche ihrer Site vom Robot nicht besucht werden soll.

Proxy-Server (Proxy Server)
Ein Server, der als Mittler für HTTP-Webanforderungen von einer Anwendung oder von einem Web-Server fungiert. Ein Proxy-Server wird als Ersatzsystem für die Content-Server im Unternehmen verwendet.

Quick Link
Eine Zuordnung zwischen einer URI und Schlüsselwörtern bzw. Ausdrücken.

Rangfolge (Ranking)
Die Zuordnung eines ganzzahligen Werts zu jedem Dokument in den Suchergebnissen einer Abfrage. Die Reihenfolge der Dokumente in den Suchergebnissen basiert auf der Relevanz für die Abfrage. Eine höhere Einstufung in der Rangfolge bedeutet eine größere Übereinstimmung. Siehe auch dynamische Rangfolge und statische Rangfolge.

Raum (Room)
Ein Programm, mit dem Benutzer die Möglichkeit haben, von anderen Benutzern zu lesende Dokumente zu erstellen, auf Kommentare anderer Personen zu antworten und den Status sowie den Endtermin eines Projekts anzuzeigen. Außerdem können die Benutzer mit anderen im selben Raum befindlichen Personen chatten. Siehe auch Lotus QuickPlace-Raum.

Reduktion auf Grundform (Lemmatization)
Der Prozess, mit dem das Lemma eines Worts in einem Wörterverzeichnis lokalisiert wird. Die Lemmatisierung unterscheidet sich von der Stammbildung darin, dass die Stammbildung algorithmisch ist und in der Regel nicht auf ein Verzeichnis zurückgreift, in dem die Wörter einer Sprache aufgelistet sind.

Regelbasierte Kategorie (Rule-Based Category)
Durch Regeln erstellte Kategorien, die angeben, welche Dokumente welchen Kategorien zugeordnet werden. Sie können beispielsweise Regeln

definieren, mit denen Dokumente, die bestimmte Wörter enthalten oder nicht enthalten oder die einem URI-Muster entsprechen, bestimmten Kategorien zugeordnet werden.

Schlüsselspeicherdatei (Keystore File)
Eine Schlüsseldatei mit öffentlichen Schlüsseln, die als Unterzeichnerzertifikat gespeichert werden, und mit privaten Schlüsseln, die als persönliches Zertifikat gespeichert werden.

Secure Sockets Layer (SSL)
Ein Sicherheitsprotokoll zur Gewährleistung von Datenschutz bei der Kommunikation.

Segmentierung (Segmentation)
Ein Prozess, bei dem die Pfadsteuerung Nachrichtenelemente (Basic Information Units) in kleinere Einheiten, so genannte BIU-Segmente aufteilt, damit diese auch von kleineren Puffergrößen in benachbarten Servern verarbeitet werden können.

Seite mit detaillierten Fehlerhinweisen (Soft Error Page)
Eine spezielle Seite, die eine detaillierte Erläuterung zu einem Fehler liefert, wenn ein HTTP-Server die von einem Client angeforderte Seite nicht zurückgeben kann, und den HTTP-Server so konfiguriert, dass dieser diese Seiten an Stelle einer Antwort zurückgibt, die lediglich aus einem Header mit einem Rückkehrcode besteht, der den Fehler näher angibt.

Servlet
Ein Java-Programm, das auf einem Web-Server ausgeführt wird und die Funktionalität des Servers erweitert, indem es auf Grund von Web-Client-Anforderungen dynamischen Inhalt generiert. Servlets werden gewöhnlich verwendet, für Datenbanken eine Verbindung zum Web herzustellen.

Sicherheitstoken (Security Token)
Informationen zu Identität und Sicherheit, mit denen der Zugriff auf Dokumente in einer Objektgruppe berechtigt wird. Verschiedene Datenquellentypen unterstützen verschiedene Sicherheitstokentypen. Beispiele: Benutzerrollen, Benutzer-IDs, Gruppen-IDs und andere Informationen für die Datenzugriffssteuerung.

Sprache XPath (XML Path) (XML Path Language (XPath))
Eine Sprache, die Teile eines XML-Quellendokuments eindeutig angibt oder adressiert. XPath bietet außerdem Basisfunktionen zur Bearbeitung von Zeichenfolgen, Zahlen und Booleschen Operatoren.

Spracherkennung (Language Identification)
Eine Funktion der Unternehmenssuche, die die Sprache eines Dokuments bestimmt.

Stammbildung (Stemming)
Siehe Wortstammbildung.

Statische Rangfolge (Static Ranking)
Ein Rangfolgetyp, bei dem Faktoren der eingestuften Dokumente (z. B. das Datum, die Anzahl der Links, die auf das Dokument verweisen, usw.) den Rang erhöhen. Vergleiche dynamische Rangfolge.

Statische Zusammenfassung (Static Summarization)
Ein Zusammenfassungstyp, bei dem die Suchergebnisse eine angegebene, gespeicherte Zusammenfassung aus dem Dokument enthalten. Vergleiche dynamische Zusammenfassung.

Steuerkomponente für Textanalyse (Text Analysis Engine)
Eine Softwarekomponente, deren Aufgabe es ist, Kontext und semantischen Inhalt in Text aufzufinden und darzustellen.

Stoppwort (Stop Word)
Ein häufig verwendetes Wort (z. B. *der*, *ein*, *und*), das von einer Suchanwendung ignoriert wird.

Stoppwortentfernung (Stop Word Removal)
Das Entfernen von Stoppwörtern aus der Abfrage, damit allgemeine Wörter ignoriert und auf diese Weise gezieltere Ergebnisse zurückgegeben werden.

Suchanwendung (Search Application)
Ein Programm, das Abfragen verarbeitet, den Index durchsucht, die Suchergebnisse zurückgibt und die Quellendokumente für Objektgruppen in einem System für die Unternehmenssuche abruft.

Suchcache (Search Cache)
Ein Puffer, der die Daten und Ergebnisse vorangegangener Suchanforderungen enthält.

Suche mit Begriffsgewichtung (Weighted Term Search)
Eine Abfrage, bei der bestimmten Begriffen größere Bedeutung beigemessen wird.

Suche nach grober Übereinstimmung (Fuzzy Search)
Eine Suche, bei der Wörter zurückgegeben werden, deren Schreibweise der des Suchbegriffs ähnlich ist.

Suchergebnisse (Search Results)
Eine Liste der Dokumente, die der Suchanforderung entsprechen.

Suchindexdateien (Search Index Files)
Gruppe von Dateien, in der ein Index in der Suchmaschine gespeichert wird.

Suchmaschine (Search Engine)
Ein Programm, das eine Suchanforderung annimmt und eine Dokumentenliste an den Benutzer zurückgibt.

Synonymverzeichnis (Synonym Dictionary)
Ein Wörterverzeichnis, das es Benutzern ermöglicht, eine Objektgruppe auch nach Synonymen ihrer Abfragebegriffe zu durchsuchen.

Taxonomie (Taxonomy)
Eine auf Ähnlichkeiten basierende Klassifikation von Objekten zu Gruppen. In der Unternehmenssuche fasst eine Taxonomie Daten zu Kategorien und Unterkategorien zusammen. Siehe auch Kategoriebaum.

Textanalyse (Text Analysis)
Das Extrahieren der Semantik und anderer Informationen aus Text, um die Abrufbarkeit von Daten in einer Objektgruppe zu verbessern.

Textbasierte Bewertung (Text-Based Scoring)
Die Zuordnung eines ganzzahligen Werts zu einem Dokument, der die Relevanz des Dokuments in Bezug auf die Abfragebegriffe anzeigt. Ein hoher ganzzahliger Wert zeigt eine große Übereinstimmung mit der Abfrage an. Siehe auch dynamische Rangfolge.

Themenextraktion (Theme Extraction)
Eine Art der Begriffsextraktion, bei der signifikante Vokabularelemente in

Textdokumenten automatisch erkannt werden, um das Thema eines Dokuments zu extrahieren. Siehe auch Begriffsextraktion.

Token Die Basistexteinheiten, die von der Unternehmenssuche indexiert werden. Tokens können aus den Wörtern einer Sprache oder aus anderen Texteinheiten bestehen, die sich für das Indexieren eignen.

Tokenizer
Ein Textsegmentierungsprogramm, das Text überprüft und ermittelt, wann und ob eine Zeichenfolge als Token erkannt werden kann.

Unicode-basierte Leerzeichensegmentierung (Unicode-Based White Space Segmentation)
Ein Aufbereitungsverfahren, bei dem mittels Unicode-Zeichenmerkmalen zwischen Token und Trennzeichen unterschieden wird.

Überwachungsbeauftragter (Monitor)
Ein Benutzer der Unternehmenssuche, der über die Berechtigung zum Beobachten von Prozessen auf Objektgruppenebene verfügt.

Uniform-Resource-Identifier (URI)
Eine kompakte Zeichenfolge, die eine abstrakte oder physische Ressource angibt.

Uniform-Resource-Locator (URL)
Eine Zeichenfolge, die Informationsquellen auf einem Computer oder in einem Netz wie dem Internet darstellt. Diese Zeichenfolge enthält den abgekürzten Namen des Protokolls, mit dem auf die Informationsquelle zugegriffen wird, sowie die Informationen, mit denen das Protokoll die Informationsquelle lokalisiert.

Universal Resource Name (URN)
Ein aus einer kurzen Zeichenfolge bestehendes Element des Internetprotokolls, das einer bestimmten Syntax folgt. Die Zeichenfolge umfasst einen Namen oder eine Adresse, mit der auf eine Ressource verwiesen werden kann.

Unstructured Information Management Architecture (UIMA)
Eine IBM Architektur, die ein Framework zur Implementierung von Systemen zur Analyse unstrukturierter Daten definiert.

Verknüpfte Suche (Proximity Search)
Eine Art der Suche, bei der nach bestimmten Wörtern im selben Satz, Absatz oder Dokument gesucht wird.

Verwaltungsrolle (Administrative Role)
Die Klassifizierung eines Benutzers, die die Funktionen festlegt, die dieser Benutzer über die Administrationskonsole für die Unternehmenssuche ausführen kann. Die Rolle legt außerdem fest, welche Objektgruppen der Benutzer verwalten kann.

Volltextindex (Full Text Index)
Eine Datenstruktur, die auf Datenelemente verweist, um einer Suche ein schnelles Auffinden von Dokumenten zu ermöglichen, die die Abfragebegriffe enthalten.

Web-Crawler (Web Crawler)
Eine Robotsoftwareklasse, die das Web durchsucht, indem sie ein Webdokument abruft und den Links in diesem Dokument folgt.

Wortstammbildung (Word Stemming)
Ein Prozess der linguistischen Normalisierung, in dem die Varianten eines

Worts auf eine allgemeine Form reduziert werden. Beispielsweise werden Wörter wie *Speicherung*, *speichernd* und *gespeichert* zu *speich-* reduziert.

Zeichennormalisierung (Character Normalization)
Ein Prozess, bei dem die Varianten eines Zeichens (z. B. Großschreibung und diakritische Zeichen) auf eine gemeinsame Form reduziert werden.

Zeilenvorschubzeichen (Newline Character)
Ein Steuerzeichen, das bewirkt, dass die Druck- oder Anzeigeposition eine Zeile nach unten versetzt wird. Bei manchen Systemen sind mehrere Zeichen erforderlich.

Zertifikat (Certificate)
Ein digitales Dokument, mit dem der Identität des Zertifikatinhabers ein öffentlicher Schlüssel angefügt wird, um eine Authentifizierung des Zertifikatinhabers zu ermöglichen. Zertifikate werden von einer Zertifizierungsstelle ausgestellt.

Zertifizierungsstelle (Certificate Authority)
Eine Organisation, die Zertifikate ausstellt und die an elektronischen Transaktionen beteiligten Entitäten (Einzelpersonen bzw. Unternehmen) authentifiziert. Zertifizierungsstellen gewährleisten, dass die beiden Informationen austauschenden Parteien, auch tatsächlich sind, wer sie vorgeben zu sein.

Zugriffssteuerungsliste (Access Control List)
Eine Liste, in der die Benutzer, die auf das zugeordnete Objekt Zugriff haben, und deren Zugriffsberechtigungen für dieses Objekt angegeben sind.

Zusammenfassung (Summarization)
Das Einfügen von Sätzen in Suchergebnisse, die den Inhalt eines Dokuments kurz beschreiben. Siehe auch dynamische Zusammenfassung und statische Zusammenfassung.

Zugreifen auf Informationen zu WebSphere Information Integration

Informationen zu WebSphere Information Integration-Produkten sind telefonisch oder über das Web verfügbar.

Die hier angegeben Telefonnummern gelten für Deutschland:

- Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.
- Unter 0180 5 5090 können Sie Handbücher telefonisch bestellen.

Informationen zu WebSphere Information Integration finden Sie auch im Web unter www.ibm.com/software/data/integration/db2ii/. Diese Site umfasst die folgenden aktuellen Informationen:

- Produktdokumentation
- Produktdownloads
- Fixpacks
- Release-Informationen und weitere Unterstützungsdokumentation
- Neuerungen zu WebSphere Information Integration
- Links zu Webressourcen wie White Papers und IBM Redbooks
- Links zu Newsgroups und Benutzergruppen
- Links zu Onlineinformationszentralen für WebSphere Information Integration-Produkte
- Bestellen von Handbüchern

Gehen Sie für den Zugriff auf Produktdokumentation wie folgt vor:

1. Rufen Sie die Website unter www.ibm.com/software/data/integration/db2ii/ auf.
2. Wählen Sie ein Produkt aus der Dropdown-Liste aus, zum Beispiel WebSphere Information Integrator OmniFind Edition.
3. Klicken Sie den Link **Support** links auf der Seite an.
4. Wählen Sie im Abschnitt **Learn** den gewünschten Link aus. Steht für das ausgewählte Produkt eine Informationszentrale zur Verfügung, können Sie den Link für die Informationszentrale auswählen. (Beispiel siehe Abb. 1 auf Seite 92)

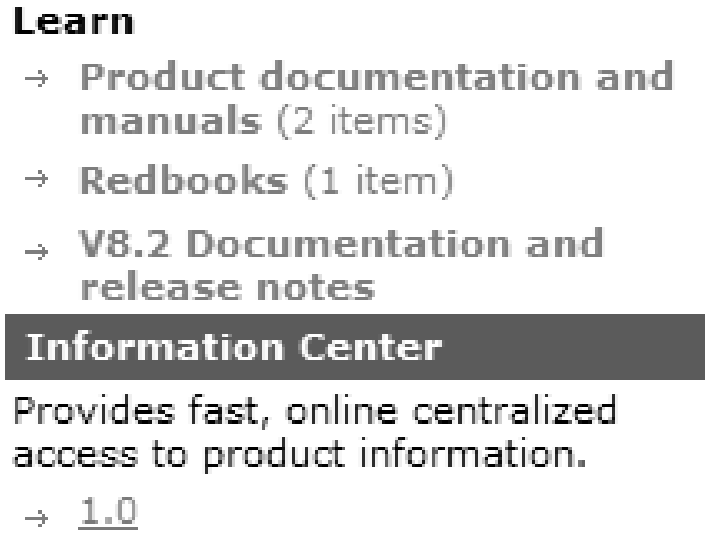

Abbildung 1. Beispiel für Links zur Produktdokumentation auf einer WebSphere Information Integration-Unterstützungswebsite

Kommentare zur Dokumentation

Bitte senden Sie uns Ihre Kommentare zu diesen Informationen oder zu anderer Dokumentation von IBM WebSphere Information Integration.

Ihre Rückmeldung unterstützt IBM, hochwertige Informationen anzubieten. Bitte senden Sie uns Ihre Kommentare zu diesen Informationen oder zu anderer Dokumentation von WebSphere Information Integration. Zum Senden von Kommentaren stehen Ihnen die folgenden Möglichkeiten zur Verfügung:

1. Senden Sie Ihre Kommentare mit Hilfe des Kommentarformulars für Onlinedokumentation unter www.ibm.com/software/awdtools/rcf/.
2. Senden Sie Ihre Kommentare als E-Mail an comments@us.ibm.com. Geben Sie den Namen des Produkts, die Versionsnummer des Produkts sowie den Namen und die Teilenummer der Informationen (falls vorhanden) an. Wenn Sie Kommentare zu bestimmtem Text haben, geben Sie die Position des Texts (z. B. einen Titel, eine Tabellennummer oder eine Seitenzahl) an.

Kontaktaufnahme mit IBM

Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.

Telefonische Unterstützung erhalten Sie über folgende Nummern:

- Unter 0180 3 313233 erreichen Sie Hallo IBM, wo Sie Antworten zu allgemeinen Fragen erhalten.
- Unter 0190 7 72243 erreichen Sie die DB2 Helpline, wo Sie Antworten zu DB2-spezifischen Problemen erhalten.

Informationen zur nächsten IBM Niederlassung in Ihrem Land oder Ihrer Region finden Sie im IBM Verzeichnis für weltweite Kontakte, das Sie im Web unter www.ibm.com/planetwide abrufen können.

Marken

In diesem Abschnitt werden IBM Marken und bestimmte Marken anderer Hersteller aufgelistet.

Informationen zu IBM Marken finden Sie in
http://www.ibm.com/legal/copytrade.shtml.

Die folgenden Begriffe sind Marken oder eingetragene Marken anderer Unternehmen:

Java und alle Java-basierten Marken und Logos sind in gewissen Ländern Marken oder eingetragene Marken von Sun Microsystems, Inc.

Microsoft, Windows, Windows NT und das Windows-Logo sind in gewissen Ländern Marken der Microsoft Corporation.

Intel, Intel Inside (Logos), MMX und Pentium sind in gewissen Ländern Marken der Intel Corporation.

UNIX ist in gewissen Ländern eine eingetragene Marke von The Open Group.

Linux ist in gewissen Ländern eine Marke von Linus Torvalds.

Andere Namen von Unternehmen, Produkten oder Services können Marken oder Servicemarken anderer Unternehmen sein.

Bemerkungen

Diese Informationen wurden für Produkte und Services entwickelt, die in Deutschland angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen nicht in allen Ländern an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. An Stelle der IBM Produkte, Programme oder Services können auch andere ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden): IBM Europe, Director of Licensing, 92066 Paris La Defense Cedex, France

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekannt gegeben. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt; die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Alle in diesem Dokument enthaltenen Leistungsdaten stammen aus einer gesteuerten Umgebung. Die Ergebnisse, die in anderen Betriebsumgebungen erzielt werden, können daher erheblich von den hier erzielten Ergebnissen abweichen. Einige Daten stammen möglicherweise von Systemen, deren Entwicklung noch nicht abgeschlossen ist. Eine Gewährleistung, dass diese Daten auch in allgemein verfügbaren Systemen erzielt werden, kann nicht gegeben werden. Darüber hinaus wurden einige Daten unter Umständen durch Extrapolation berechnet. Die tatsächlichen Ergebnisse können abweichen. Benutzer dieses Dokuments sollten die entsprechenden Daten in ihrer spezifischen Umgebung prüfen.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichen Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Die oben genannten Erklärungen bezüglich der Produktstrategien und Absichtserklärungen von IBM stellen die gegenwärtige Absicht der IBM dar, unterliegen Änderungen oder können zurückgenommen werden, und repräsentieren nur die Ziele der IBM.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufes. Sie sollen nur die Funktionen des Lizenzprogrammes illustrieren; sie können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden; Ähnlichkeiten mit tatsächlichen Namen und Adressen sind rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Musteranwendungsprogramme, die in Quellensprache geschrieben sind. Sie dürfen diese Musterprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, verwenden, vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle konform sind, für die diese Musterprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten.

Kopien oder Teile der Musterprogramme bzw. daraus abgeleiteter Code müssen folgenden Copyrightvermerk beinhalten:

Outside In (®) Viewer Technology, © 1992-2005 Stellent, Chicago, IL., Inc. Alle Rechte vorbehalten.

IBM XSLT-Prozessor Lizenziertes Material - Eigentum der IBM © Copyright IBM Corporation, 1999-2005. Alle Rechte vorbehalten.

Index

W

Z

SC12-3611-00